A Knowledge Integration Framework for 3D Shape Reconstruction

Thesis

A Knowledge Integration Framework for 3D Shape Reconstruction

Eugen Funk

Dipl. Ing. (M. Sc.)

A thesis submitted in partial fulfilment of the requirements for the degree of

Doctor of Philosophy

Department of Computing and Communications

Faculty of Mathematics, Computing and Technology

The Open University

Milton Keynes, United Kingdom

Submitted on 7th March 2016

Abstract

The modern emergence of automation in many industries has given impetus to extensive research into mobile robotics. Novel perception technologies now enable cars to drive autonomously, tractors to till a field automatically and underwater robots to construct pipelines. An essential requirement to facilitate both perception and autonomous navigation is the analysis of the 3D environment using sensors like laser scanners or stereo cameras. 3D sensors generate a very large number of 3D data points in sampling object shapes within an environment, but crucially do not provide any intrinsic information about the environment in which the robots operate with. This means unstructured 3D samples must be processed by application-specific models to enable a robot, for instance, to detect and identify objects and infer the scene geometry for path-planning more efficiently than by using raw 3D data.

This thesis specifically focuses on the fundamental task of 3D shape reconstruction and modelling by presenting a new knowledge integration framework for unstructured 3D samples. The novelty lies in the representation of surfaces by algebraic functions with limited support, which enables the extraction of smooth consistent shapes from noisy samples with a heterogeneous density. Moreover, many surfaces in urban environments can reasonably be assumed to be planar, and the framework exploits this knowledge to enable effective noise suppression without loss of detail. This is achieved by using a convex optimization technique which has linear computational complexity. Thus is much more efficient than existing solutions.

The new framework has been validated by critical experimental analysis and evaluation and has been shown to increase the accuracy of the reconstructed shape significantly compared to state-of-the-art methods. Applying this new knowledge integration framework means that less accurate, low-cost 3D sensors can be employed without sacrificing the high demands that 3D perception must achieve. This links well into the area of robotic inspection, as for example regarding small drones that use inaccurate and lightweight image sensors.

I dedicate this work to...

My son Karl and my wife Franziska. Thank you for your endless support, understanding and love.

My parents, who taught me to analyse things critically.

Acknowledgement

Most of all I would like to thank my main supervisor Prof. Laurence S. Dooley, who improved my academic writing with his patience, guidance and inexhaustible support.

Anko Börner, my second supervisor from the German Aerospace Center, who turned my PhD odyssey into a fruitful journey and always kept supporting me by all means.

My friends Marc, Jens, Lisa and Alex for all their inspiration, motivation by outstanding PhD examples and great friendship.

I also would like to thank Prof. Heinz-Wilhelm Hübers for funding and the great opportunity to participate in the Helmholtz Research School on Security Technologies, which broadened my view on security research.

Prof. A. Paulton and Prof. O. Hellwich for their guidance and motivation.

Parminder and Smarti Reel, my friends based in Milton Keynes, who have supported me for the past 5 years including a friendly work environment.

Declaration

The work presented in this thesis is an original contribution of the author. Parts of this thesis have appeared in the following

Peer-Reviewed Publications:

1. **Funk E.**, Griessbach D., Baumbach D., Ernst I., Börner A., Zuev (2011). *Segmentation of large point-clouds using recursive local PCA.* Poster In International IEEE conference on Indoor Positioning and Navigation (IPIN), Guimarães, Portugal

2. **Funk, E.**, Dooley, L. S., Zuev, S., Börner, A. (2012). *Visual recognition of scenes and objects on trains using stereo cameras* . In European Navigation Conference, Gdansk, Poland

3. Grießbach, D., Baumbach, D., Börner, A., Buder, M., Ernst, I., **Funk, E.**, Wohlfeil, J., Zuev, S. (2012). *IPS – A system for real-time navigation and 3D modeling.* In The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XXXIX (B3), pp. 75-80. ISPRS. XXIInd ISPRS Congress, 25. Aug. - 01. Sep. 2012, Melbourne, Australien. *DOI:* $10.5194/isprsarchives - XXXIX - B3 - 75 - 2012$

4. **Funk, E.**, Dooley, L. S., and Börner, A., Grießbach, D. (2013). *Implicit scene modelling from imprecise point clouds.* In ISPRS Commission IV/7: Conference on Acquisition and Modelling of Indoor and Enclosed Environments, XL4/ (W4-7), 11.–13. Dez. 2013, Cape Town, South Africa. *DOI: 10.5194/isprsarchives-XL-4-W4-7-2013*

5. **Funk, E.**, Dooley, L., Börner A. (2015). *TVL_1 Shape Approximation from Scattered 3D Data.* In Computer Vision, Imaging and Computer Graphics. Theory and Applications, 10th International Joint Conference, VISIGRAPP 2015, Berlin, Germany, March 11-14, 2015, Selected Papers, *ISBN: 978-3-319-29971-6.*

6. **Funk, E.**, Dooley, L., Börner A. (2016). *TVL_1 Planarity Regularization for 3D Shape Approximation.* In Communications in Computer and Information Science, Springer International Publishing, *DOI:* $10.1007/978 - 3 - 319 - 29971 - 6_15$.

Eugen Funk

Submitted on 7^{th} March, 2016

List of Abbreviations

ADMM	Alternate Direction of Multipliers
CPU	Central Processing Unit
CSRBF	Compactly Supported RBF
CV	Cross Validation
GCV	Generalized Cross Validation
GPL	General Public License
GUI	Graphical User Interface
IIR	Integrated Iterative Research
IPS	Integrated Positioning System
L_1	Vector norm $\| \boldsymbol{\alpha} \|_1 = \sum \lvert\alpha_i\rvert$
L_2	Vector norm $\| \boldsymbol{\alpha} \|_2^2 = \sum \alpha_i^2$
Lasso	Least Absolute Shrinkage and Selection
LSQ	Least Squares
MDE	Mean Distance Error

MLS	Moving Least Squares
MSE	Mean Square Error
NURBS	Non-Uniform Rational B-Spline
PDE	Partial Differential Equation
RBF	Radial Basis Function
RIMLS	Robust Implicit Moving Least Squares
SfM	Structure from Motion
SGM	Semi Global Matching
SOR	Successive Over Relaxation
SSDF	Smooth Signed Distance Function
TPS	Thin Plate Spline
TV	Total Variation
TVL_1	Total Variation minimization using L_1 norm
TVL_2	Total Variation minimization using L_2 norm
$\mathbf{x}$	Location in 3D space
$\mathbf{x}_i$	i^{th} sample point in 3D space
$\boldsymbol{\alpha}$	Vector of RBF weights
φ	Radial basis function, also known as kernel function

Contents

List of Figures

List of Tables

1 Introduction

The importance of mobile robotics in many industries and in our everyday lives has increased significantly in the last decade. Research on autonomous robotics allows many handcraft tasks to be automated. Today, supportive driving on motorways, automated farm tractors or pallet trucks in warehouses are practical to a certain extent. A critical task of all autonomous systems is the perception of the surrounding environment. Perception depends on the analysis of data from 3D sensors such as laser scanners or stereo cameras. In fact, the key to most applications related to mobile robotics is the ability to reconstruct and analyse a 3D shape model of the environment. Thus, technologies integrating 3D shape reconstruction have the potential for a significant breakthrough in the future, in terms of both social and economical impact (Hägele, 2011). When modelling the environment from 3D sensor data, automated decisions like, for example, how to pick up an object, how to avoid a collision or how to recognize objects become feasible.

1.1 Stereoscopic Cameras for 3D Reconstruction

State-of-the-art mobile 3D perception systems rely mainly on large 3D laser scanners to sample the geometry of the environment by 3D points as shown in Figures 1.1a and

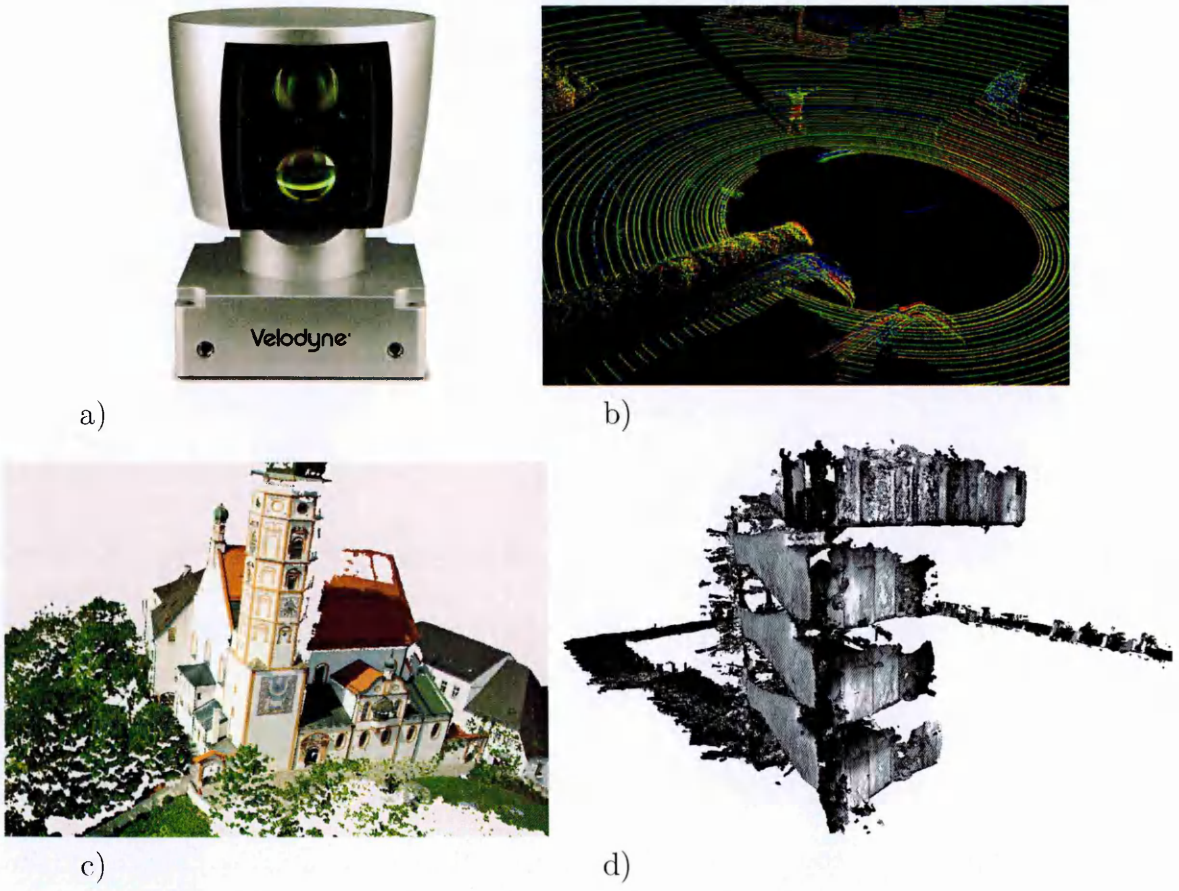
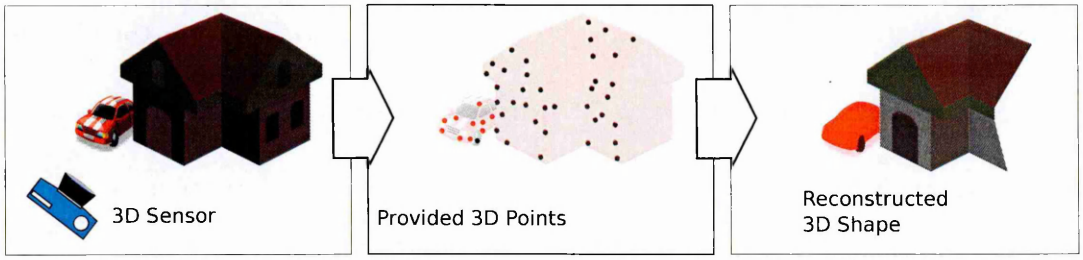


Figure 1.1: a) Modern 3D scanner (LiDAR) from Velodyne, b) 3D point cloud from a street, c) 3D point cloud of a chapel obtained via time consuming post processing (SfM), d) Error prone 3D points of a building interior obtained in real-time from a mobile stereoscopic camera system.

1.1b. However, these sensors are expensive and their application on mobile platforms can be cumbersome due to their weight and power consumption. For this reason, camera-based 3D shape reconstruction has become a major focus of research. However, mobile platforms are often required to perform the 3D reconstruction process in real time, which prevents the application of high-accuracy, but computationally intensive techniques such as the *Structure from Motion* (SfM) algorithm. Furthermore, the 3D data obtained from mobile stereo platforms often contains many errors and outliers, as illustrated in Figures 1.1c and 1.1d. Collectively, these issues mean the quality of the 3D point data makes it extremely challenging to either create suitable 3D models or to



a) b) c)

Figure 1.2: a) 3D sensor observing a scene, b) obtained 3D samples, c) 3D shape reconstructed from the samples.

perform semantic scene analysis in order for a robot to automatically interpret what is in front of the camera. For these reasons, improving the 3D data quality is a key aim in this thesis. The main process of automated 3D modelling is further illustrated in Figure 1.2 where 3D sensor samples are used as an input.

First, the 3D sensor (camera icon) observes the objects (building and vehicle) in Figure 1.2a and generates 3D samples (shown in Figure 1.2b) of the object geometries. Each 3D sample is represented by spatial coordinates (x, y, z) together with a colour value. The final step is then the construction of the 3D model (Figure 1.2c using the 3D samples. A crucial feature of this final step is that it is expected to be able to ignore error-prone 3D samples at wrong locations and to represent the walls, the roof and the car as accurate as possible.

1.2 Data Processing Challenges and the Research Question

Being able to improve the quality of a 3D model relying on error-prone 3D samples would enable a large set of applications with mobile camera devices. This motivated several research communities to investigate the noise and outliers reduction techniques in 3D point data. As a starting point, triangulation techniques have been applied widely.

However, these directly interconnect the acquired 3D samples to represent the shapes by triangle meshes (Cazals and Giesen, 2004) which introduces errors in the 3D shape model directly from the sample data. Furthermore, when the spatial distribution of the 3D samples is non-uniform and the distance between samples is random (scattered), the triangulation process produces triangles at random positions with random orientations.

In order to reduce the effect of noise in the raw 3D samples, the research community has investigated the statistical filtering of samples (Huber et al., 1981). While smoothing or outlier removal techniques have become state of the art (Harrell, 2015), they involve the removal of important information by processing the data. New statistical shape modelling strategies are therefore required to effectively combine both outlier removal and detail-aware filtering without crucially removing key information from the input 3D dataset. In contrast to standard triangulation techniques where data samples are interpolated, new techniques need to be developed which can approximate the data while having an awareness of the errors in particular 3D samples. This provided the motivation to explore how *a priori* knowledge can be exploited effectively to address this challenge and leads to the overarching **research question** of the thesis:

> *How can scenes be modelled by using 3D shape approximation from scattered 3D samples?*

The research question is addressed by the new knowledge integration framework shown in Figure 1.3. Starting with error-prone 3D point clouds, the framework applies several novel approaches in order to generate a high accuracy 3D shape. Referring to the process illustration in Figure 1.2, the novel framework lieas in between steps 2 and 3. The framework consists of three key elements. Firstly, a new mathematical representation

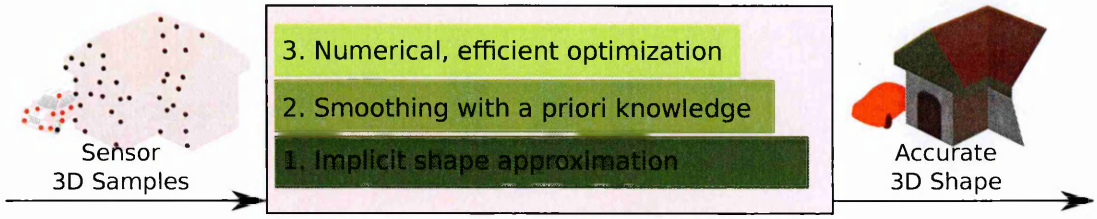


Figure 1.3: The proposed knowledge integration framework.

model is formulated that facilitates the approximation of scattered 3D samples. A new implicit shape model is proposed which enables the extrapolation of missing data (holes) and the representation of shapes with arbitrary smoothness. Regarding outlier removal, research (Wekel and Hellwich, 2015) has shown that when exploiting additional environment-specific information, it is possible to improve the quality of the model extraction significantly. In the second aspect of the framework, the proposed implicit shape formulation is extended to enable the integration of environment-specific information, that is *a priori* knowledge, into the 3D shape approximation process. Finally, in order to be able to process realistic datasets comprising millions of 3D samples, the numerical optimization algorithm is extended to linear complexity without compromising the accuracy of the final shape approximation results. In the next section these three aspects will be described in detail.

1.3 Research Objectives and Hypotheses

Research Objective 1:

To develop and critically evaluate new kernel-based models for geometry approximation from scattered 3D samples.

The task of 3D shape reconstruction from scattered points has attracted much attention in the computer graphics community (Alexa et al., 2001; Kazhdan and Hoppe, 2013). In contrast to robotics applications, which is the main motivation of this thesis, the goal is to support manual modelling of movie characters or film sceneries. That means the focus of research has been limited to accurate dense 3D point clouds that have been acquired by stationary 3D scanning systems in controlled environments (Bodenmüller, 2009).

In recent times, 3D modelling has focused on the 3D samples generation step as shown in Figure 1.2b with significant improvements in accuracy being reported (Wei et al., 2014). Unfortunately, the computational overheads are extremely high, with accurately computed 3D samples from camera images taking several days. Thus, lower accuracy methods are favoured when results in real time are required. This leads to larger errors in the 3D measurements, which prevents the application of state-of-the-art shape reconstruction methods.

Moreover, in unknown environments the accuracy of 3D sensors decreases dramatically (Meister et al., 2012) even if time consuming off-line 3D sample reconstruction techniques are applied. Difficult illumination conditions cause outliers, homogeneous colours of surfaces lead to noise, and occlusions make accurate 3D shape reconstruction even more challenging. Figure 1.4 shows an example of data acquired from a mobile 3D sensor. Strong noise and missing samples due to occlusion and reflections are evident. Dealing with noise and outliers naturally involves techniques from the computational statistics domain (Bishop, 2006). In the past decade, kernel-based optimization techniques have become important because they can significantly enhance the approximation accuracy of

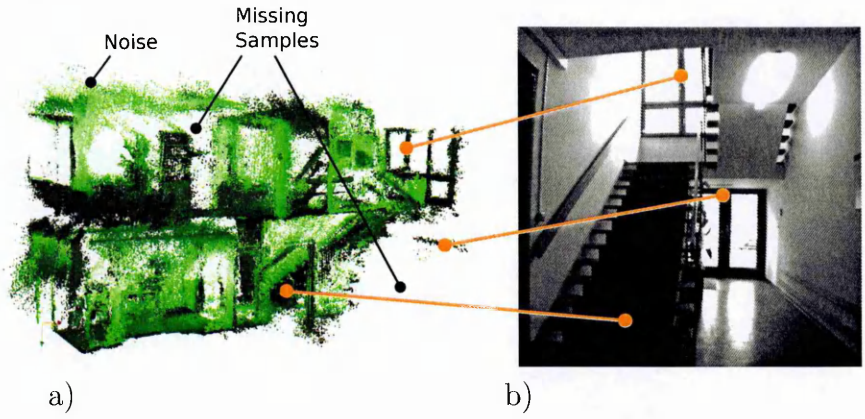


Figure 1.4: a) The acquired 3D samples shown as a point cloud. b) A part of the scanned stair house as a photograph.

machine learning applications (Scholkopf and Smola, 2001) and statistical data analysis in general (Fasshauer, 2011). This provided the motivation for the research objective to investigate and evaluate critically how kernel-based approximations can be applied effectively to 3D shape reconstruction. The goal is to enable consistent surfaces of arbitrary smoothness and the extrapolation of missing samples.

Research Objective 2:

> *To develop models which exploit planarity information, support noise suppression and improve shape approximation accuracy in error-prone 3D datasets.*

The integration of *a priori* knowledge into the model extraction process has been proven to increase accuracy (Wang et al., 2011). In many application domains such as indoor or urban outdoor environments most shapes are planar in nature. Bredies et al. (2010) proposed an edge-aware minimization strategy for the number of corners and edges in depth images by applying *total variation* (TV) minimization. Each pixel in a depth image encodes the distance of the object away from the sensor as presented in Figure 1.5, which shows a single object (building). Bredies et al. (2010) compared the TV

filtering approach to several state-of-the-art estimators such as averaging (Figure 1.5c), median (Figure 1.5d and the Huber norm (Huber et al., 1981) (Figure 1.5e). Since TV minimization assumes that most height measurements lie on a plane, the building model extracted from the noisy input data can be very accurate. Similar challenges arise in 3D shape approximation, since the 3D samples usually contain significant errors. This means that integrating planarity regularization into the proposed framework represents an attractive opportunity to pursue in seeking superior 3D shape reconstruction accuracy.

Research Objective 3:

Critically synthesize a new computationally tractable TV 3D shape approximation technique.

Fulfilling research objective 2 by minimizing the TV in combination with the 3D shape reconstruction process, has a major impact on the overall computational complexity. When the number of 3D samples is increased, the order of computational time increases by an order of three when standard numerical techniques are applied. This prohibits their application on realistic datasets that typically comprise millions of points. Thus, new

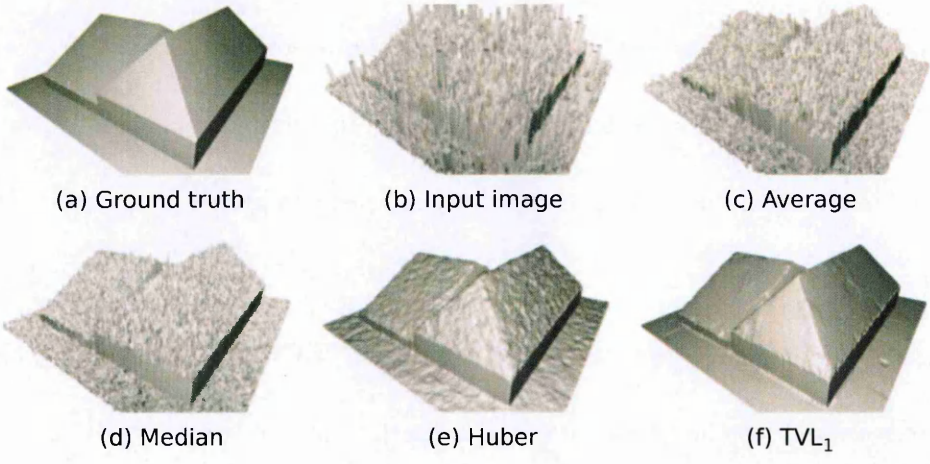


Figure 1.5: Edge-aware image filtering via TV minimization (Bredies et al., 2010).

more efficient algorithms need to be designed and developed which are computationally faster but do not compromise the accuracy and noise suppression performance while at the same time maintain the shape approximation smoothness.

The underlying work consolidates all three research objectives in an innovative framework shown in Figure 1.3 for an efficient and reliable 3D shape approximation technique for error-prone scattered 3D points. The integration of smoothness information as *a priori* knowledge is a fundamental contribution which, by means of an efficient iterative optimization algorithm, extends the implicit shape model representation. A summary of the main contributions of this research will be presented now.

1.4 Contributions

The new knowledge integration framework for 3D shape reconstruction proposed in this thesis makes three original contributions. The first is the development of a novel automated shape approximation technique using an implicit non-parametric function.

The second contribution integrates the planarity assumption on the approximated geometry as *a priori* knowledge into the approximation process. It is assumed that the second derivative of the implicit function is zero when the shape is planar. This is exploited for data-adaptive shape smoothing while being aware of the corners and edges sampled by 3D points. Thus, the second derivative of the shape is jointly minimized during the reconstruction process and the number of edges and corners is reduced.

The third contribution relates to the numerical optimization of the techniques developed in the first two framework contributions. Common TV minimization techniques suffer from high complexity of $\mathcal{O}(N^3)$ (Boyd and Vandenberghe, 2004). As it will be

analysed critically in Chapter 6, the new technique incurs linear $\mathcal{O}(N)$ complexity without compromising the accuracy, noise suppression performance and the approximation of piecewise smooth shapes.

1.5 Structure of the Thesis

The remainder of the thesis is organized as follows:

- Chapter 2 provides a comprehensive literature review of contemporary 3D geometry representation and approximation techniques. As will be shown, modern approaches focus on the integration of specific *a priori* knowledge into the shape reconstruction process in order to increase the modelling accuracy. The critique in Chapter 2 especially emphasizes key aspects of existing methods including their efficiency, robustness to noise, the ability to deal with sparse measurements and overall accuracy.

- Chapter 3 describes the applied integrated iterative research methodology adopted in this thesis. This methodology integrates aspects from the literature research, idea prototyping, testing and validation. In addition, both qualitative and quantitative evaluation strategies are presented including details of the test simulation platform, performance metrics and test datasets used for benchmarking.

- Chapter 4 introduces the theory of non-parametric (kernel-based) approximations. As this is fundamental to the development of the new implicit shape model, radial basis functions RBF are critically reviewed with respect to the regression stability from redundant and sparse samples. The effect of different regularization techniques is illustrated with simple examples which allow generalization and direct application on the 3D shape approximation from 3D samples. Chapter 4 concludes with a series of generalized guiding principles for selecting a specific RBF type for a 3D shape approximation task.

- Chapter 5 presents the first contribution of this work by proposing a kernel-based (RBF) shape reconstruction approach for 3D samples. The shape planarity assumption highlighted above in research objective 2 is enforced by minimizing the second derivatives of the shape. Its effect is demonstrated on synthetic geometrical models with different noise levels. It is shown that the proposed technique enables significant improvements in the approximation quality from scattered and error-prone 3D samples. Parts of this chapter have been published in Funk et al. (2014).

- Chapter 6 introduces an efficient numerical solution to the 3D shape reconstruction technique developed in Chapter 5. The originality of this work is the development of an iterative optimization method capable of efficiently solving the TVL_1 shape approximation, so reducing the order of computational complexity from cubic to linear without affecting the quality of the shape reconstruction. Parts of this chapter have been published in Funk and Börner (2014), Funk et al. (2015) and Funk et al. (2016).

- Chapter 7 presents an evaluation of the new knowledge integration framework on indoor and outdoor scenes. Critical qualitative and quantitative analysis underpins the improved quality and robustness to existing shape reconstruction approaches. The error metric, which is the statistical difference between the ground truth data and the reconstructed shapes, is reduced by approximately 20% when the proposed technique is applied.

- Chapter 8 discusses some potential research directions which can exploit the new

knowledge integration framework. This can be expected to have a significant impact on the domain of automated 3D modelling.

- Chapter 9 provides some conclusions on the findings presented in this thesis.

1.6 Summary

This chapter has presented the context for the development of a new 3D shape approximation framework with the novelty being the integration of *a priori* knowledge into the shape reconstruction process. The focus is especially on planarity-aware shape approximation which enables enhanced accuracy when surfaces in either urban or indoor environments are approximated. Three research objectives have been framed which specifically relate to the development of a formal shape representation methodology, a strategy for integrating *a priori* planarity knowledge and the efficient computation strategy for very large 3D datasets.

As the first step to achieving these objectives, the next chapter presents a critical literature review of existing knowledge integration approaches to 2D and 3D shape approximation. Issues addressed in the contributions such as shape smoothness and robustness to error in the data are particularly highlighted.

2 Review of 3D Shape Modelling

This chapter will review the current state of the art in shape reconstruction from unordered 3D points. Initially, general shape representation techniques are reviewed which are suitable for numerical shape modelling and approximation. Thereafter, modern approaches that integrate *a priori* knowledge into the shape reconstruction process are evaluated. The most promising approaches are discussed with respect to the research goals of this work.

Shape reconstruction and approximation has been addressed by the computer graphics community for the past two decades. Initially, direct triangulation techniques have been developed (Edelsbrunner and Mücke, 1994), which, however, are not applicable on scattered 3D points because of noise and redundancy. The difficulty has later been elaborated by integrating smoothness *a priori* knowledge, also called *prior*, into the approximation process (Alexa et al., 2001; Calakli and Taubin, 2011; Kazhdan and Hoppe, 2013). This allows the reconstruction of smooth surfaces even if the samples are nearly redundant and distributed randomly. This important property is further extended in the presented research. However, a serious issue remains: the suppression of noise. The smoothness prior integration motivated many research groups to develop new application-specific *a priori* models. Some of these consider repetitive structures (Pauly et al., 2008; Berner et al., 2011) while others consider the geometry of the sampled scene

(Bao et al., 2013). Capabilities of such approaches are further reviewed in this chapter. Also smoothness priors have been extended to global and piecewise smooth geometries (Boulch et al., 2014; Avron et al., 2010; Bredies et al., 2010). Since smoothness is the most common feature in many applications, this will be investigated in more detail as it forms the basis of research objective 2. The discussion is then summarized in Section 2.3, where robustness, computational effort, handling of sparse measurements and the overall accuracy of existing methods are elaborated.

2.1 General Shape Representation

2.1.1 Representation via Simplexes

In the domain of computer graphics shape representation via polygon meshes, or more generally with simplexes, is an accepted practise. This is mainly driven by interactive visualization applications such as games and virtual reality simulations (Hughes et al., 2014). The research in computer graphics leads to a large number of rendering frameworks such as OpenGL (Wolff, 2013) and enables the visualization of simplical polygon meshes with the help of parallel hardware. For this reason, early research of automated shape reconstruction from 3D points focused on the

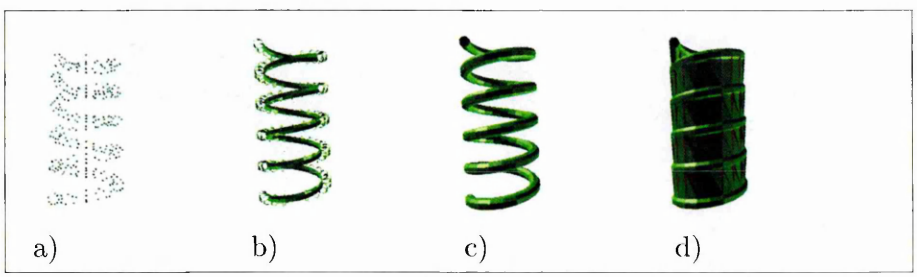


Figure 2.1: The α-*shapes* algorithm. a) Input samples, b-d) reconstruction with increasing α. Edelsbrunner and Mücke (1994).

direct construction of triangle meshes connecting tetrahedra via Delaunay-triangulation (Bowyer, 1981; Edelsbrunner, 1987). Edelsbrunner and Mücke (1994) and Bajaj et al. (1995) proposed the so-called α-shapes (Figure 2.1) for creating a closed and topologically correct surface. The method aims at connecting neighbouring points via triangles while the α value controls the acceptable Euclidean distance between the connected samples. Since α is defined by an expert user, the method is not practical for non-interactive applications. Later, Bernardini et al. (1999) proposed a more adaptive region growing technique called *Ball-Pivoting Algorithm* (BPA), though a major recurring drawback of α shapes, BPA and all Delaunay-triangulation-based methods is their high sensitivity to noise and outliers (Bodenmüller, 2009). The reason for this is that each sample is used to construct triangles even if it is an outlier or an error-prone measurement (Cazals and Giesen, 2004).

2.1.2 Parametric Representation

Parametric surface-fitting algorithms, which are also called spline-based methods, are well-known for signal approximation as well as for interpolation. The goal is to find a function $f(u, v) : \mathbb{R}^2 \mapsto \mathbb{R}$ that acts on a planar space $\mathbb{R}^2$ and gives the height

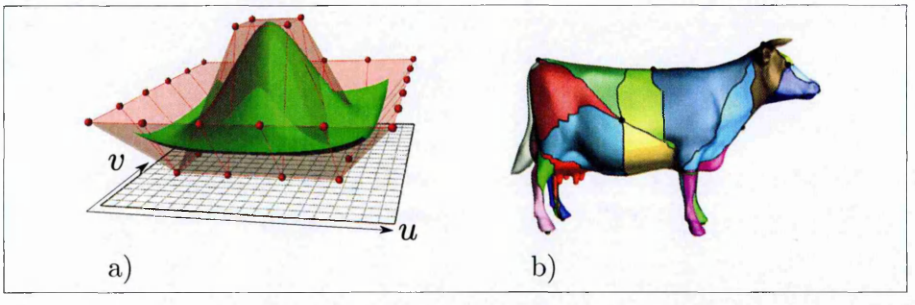


Figure 2.2: a) Smooth surface model via NURBS. b) A set of parametric shapes combined to a global consistent surface. Schreiner et al. (2004).

with respect to the plane it is approximating. Traditionally, a parametric surface reconstruction algorithm consists of two main steps:

1. **Partitioning**: Application of Delaunay-triangulation or a clustering technique (Sheffer et al., 2007),

2. **Parametrization**: For each segment, a local plane and the corresponding height parametrization model f is extracted via optimization.

A common practise is to apply the *least squares* technique to minimize

$$\min_{f} \sum_{i}^{N} \| h(\mathbf{p}_i) - f(u_i, v_i) \|_2^2$$

where $\mathbf{p}_i \in \mathbb{R}^3$ is the i^{th} input 3D sample, $h(\mathbf{p}_i) : \mathbb{R}^3 \mapsto \mathbb{R}$ is its height in a segment and $(u_i, v_i) = \text{proj}(\mathbf{p}_i)$ is the projection of $\mathbf{p}_i$ on the corresponding segment plane partition from step 1.

Several spline models such as *non-uniform rational B-Splines* (NURBS) (Piegl and Tiller, 1997; Rogers, 2001) or Bezier curves (Agoston, 2005) are commonly employed to model f of each segment. These methods are able to create smooth surfaces for non-uniform control point sets (Figure 2.2a). Locally parametrized segments are combined to give a global continuous shape (Figure 2.2b), which is, however, a combinatorial task with high computational complexity as shown by Floater and Hormann (2005). For this reason, global shape modelling with NURBS or Bezier-curves is not used. Instead local-based approximations are preferred as will be examined in greater depth in Section 2.2.2.

2.1.3 Implicit Representation

The *Signed Distance Field* (SDF) is a special case of shape representation. It is of high potential for applications such as camera movement tracking (Canelhas et al., 2013), collision detection and cloth animation (Bridson et al., 2003), multi-body dynamics (Guendelman et al., 2003) or motion planning (Hoff et al., 1999). An implicit shape is basically an indicator function $f(\mathbf{x})$ to classify the 3D space as *inside* $f(\mathbf{x}) < 0$ or *outside* $f(\mathbf{x}) > 0$ with $\mathbf{x} \in \mathbb{R}^3$ as 3D coordinate. The surface of the object is the set of all $\mathbf{x}$ where f gives zero as illustrated in Figure 2.3. The 3D space is divided into a fine grid in which each element, called *voxel*, contains the implicit indicator $f(\mathbf{v})$. Newcombe et al. (2011) demonstrated high accuracy 3D modelling with SDF by applying it to the Kinect sensors, where the data is provided as a stream.

The disadvantage of SDF is that the complete 3D space has to be divided into a large dense set of cells even for areas where no measurements have been registered. Because of this high memory requirement, the representation method is not feasible for larger areas. Approximating a volume $100 \times 100 \times 100m^3$ on a grid of 1cm would thus require 4000GB using standard 32 bit floating point values to encode the implicit SDF.

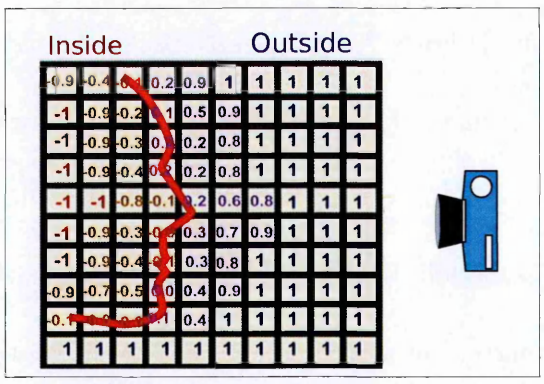


Figure 2.3: A Signed Distance Function (SDF) on a fine grid.

The general shape representations discussed so far enable the approximation of arbitrary geometries. In many applications, however, this is not required and it is preferable to restrict the generality of the representation approach in favour of the approximation quality. Examples in the next section will illustrate how more explicit *a priori* information can be exploited effectively to handle missing measurements and to approximate a shape provided the underlying model is known.

2.2 Prior-Based Shape Reconstruction

The integration of prior knowledge into the reconstruction task is essential when surfaces are required to be reconstructed from noisy data. Two general prior types have been identified: i) regular or repetitive structures, ii) smoothness and piecewise smoothness priors.

2.2.1 Regularity Priors

Pauly et al. (2008) proposed clustering point clouds into repetitive segments and exploiting this information for hole filling of structured environments as shown in Figure 2.4. Berner et al. (2011) developed a more general notion of partial symmetries, which is represented in a low dimensional shape space. A basic structure is firstly detected

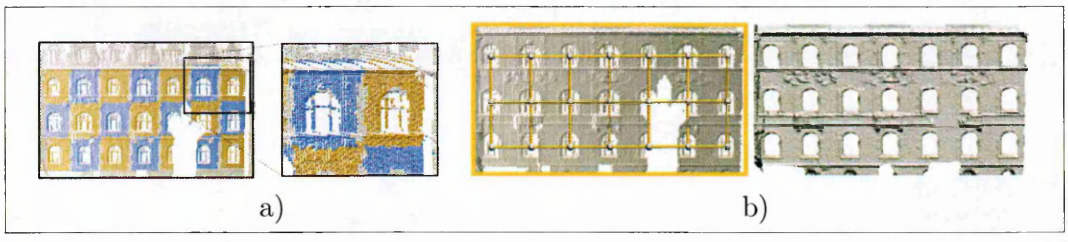


Figure 2.4: Detecting repetitive structures (a) enables hole filling (b) in structured environments. Pauly et al. (2008).

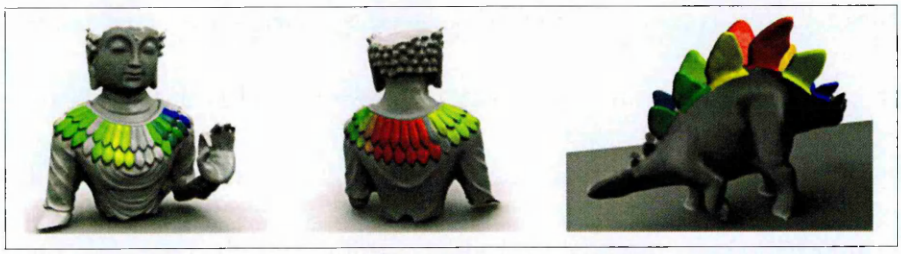


Figure 2.5: Similar objects are scaled to match repetitive patterns. Red: strong deformations. Green: small deformations. Berner et al. (2011).

and then applied to similar areas by means of non-rigid deformations. In Figure 2.5, for example, strong deformations in the shape subspace are coloured in red while small variations are shown in green. Berner et al. (2011) further proposed to apply supervised segmentation of the data. Supervised integration of priors has also been investigated by Sharf et al. (2007) and Arikan et al. (2013), where topology or relation-based similarities are marked by the user. In order to reduce the effort from interactive intervention, Bao et al. (2013) and Shen et al. (2012) perform the classification of the observed images and sparse point clouds by fitting a known model from the database (Figure 2.6) and using regularity and pre-learning approaches. These methods are expected to perform well when known objects are observed. The methods do not provide any advantage when the scene consists on unknown elements, which is a more common case. Thus, more straightforward and generalized prior models are required.

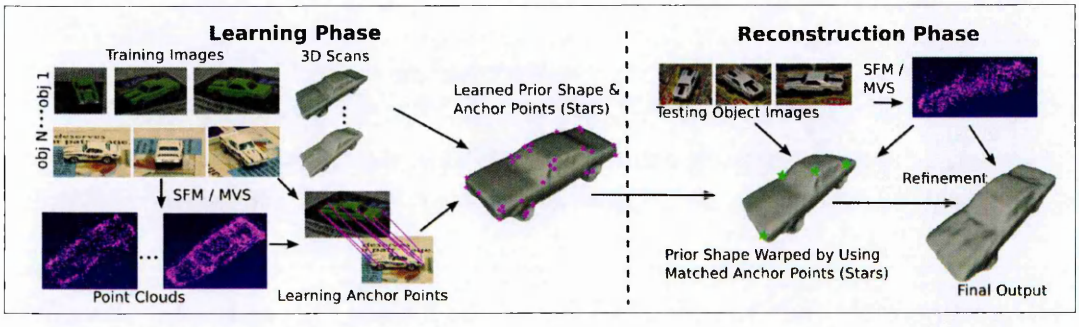


Figure 2.6: Bao et al. (2013) learn priors from images for reconstruction.

Smooth surfaces have been identified as the most common denominators for a large variety of scenes. So integrating this information into the shape reconstruction approach does not restrict an algorithm to a specific application. The smoothness assumption is divided into three categories: i) local, ii) global and iii) piecewise smoothness. Each of these will now be considered.

2.2.2 Local Smoothness Priors

Alexa et al. (2001) proposed the approximation of the implicit shape in the local neighbourhood around each sample. Since only neighbouring samples are used in the approximation process, methods of this type are known as *moving least squares* (MLS) techniques. Discontinuities are smoothed by a decreasing smooth weighting function Θ. This enables smooth shape transitions between two different approximation centres $\mathbf{x}$ as illustrated in Figure 2.7a. In the first step, a plane function $h(\mathbf{x})$ is extracted from the surrounding samples (green support in Figure 2.7a):

$$h(\mathbf{x}) = \arg\min_{\mathbf{n},d} \sum_i (\mathbf{n} \cdot \mathbf{p}_i - d)^2 \Theta(\| \mathbf{p}_i - \mathbf{x} \|_2).$$

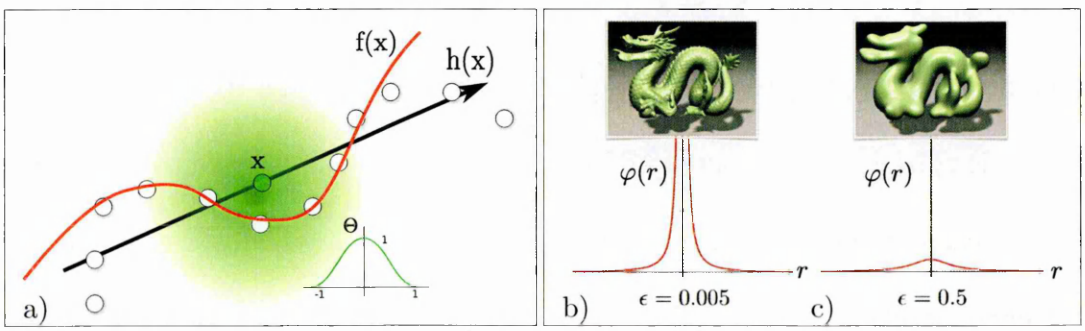


Figure 2.7: a) Local surface approximation by Alexa et al. (2001). b-c) Controlled smoothness by point-to-plane blending by Kolluri (2005).

In the second step, a smooth polynomial function $f(\mathbf{x})$ is approximated via LSQ to give the height $h(\mathbf{p}_i)$ of each sample:

$$\arg\min_f \sum_i (f(u_i, v_i) - h(\mathbf{p}_i))^2 \Theta(\| \mathbf{p}_i - \mathbf{x} \|_2).$$

Finally, the samples are re-sampled along the estimated shape and rendered as proposed by Rusinkiewicz and Levoy (2000). This approach has received much attention due to its simplicity and smooth continuous representation. Kolluri (2005) further proposed a technique enabling the control of the smoothness via point-based blending that integrates the sample normals into the shape function

$$f(\mathbf{x}) = \frac{\sum_i \mathbf{n}_i \cdot (\mathbf{p}_i - \mathbf{x}) \varphi(\| \mathbf{p}_i - \mathbf{x} \|_2)}{\sum_i \varphi(\| \mathbf{p}_i - \mathbf{x} \|_2)}$$

where the *sharpness* weighting

$$\varphi(r) = \frac{1}{r^2 + \epsilon}$$

can be controlled via the user-specified parameter ϵ. Figure 2.7b illustrates the effect of ϵ on the smoothness of the shape. Higher ϵ values lead to over-smoothing but enable more accurate reconstruction when noise is present.

Guennebaud and Gross (2007) extended the polynomial model from Alexa et al. (2001) by fitting spheres to local samples. The approach is feasible to reconstruct shapes from sparse measurements as illustrated in Figure 2.8a. However, dealing with noise remains a severe issue.

All the aforementioned local methods approximate a shape from neighbouring samples

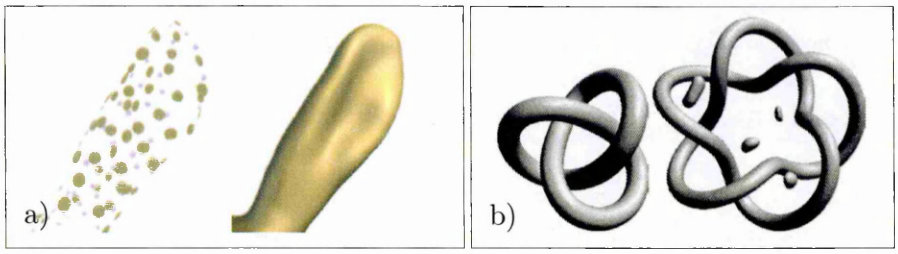


Figure 2.8: a) Algebraic sphere fitting from sparse samples. Guennebaud and Gross (2007). b) MLS without and with outliers. Ohtake et al. (2003).

valid only for the single query position $\mathbf{x}$. In the case of strong noise or outliers, these approaches fail and lead to artefacts (Figure 2.8b). In order to overcome this, a family of techniques which aim to reconstruct a shape by minimizing a global cost function is employed. These will now be reviewed.

2.2.3 Global Smoothness Priors

Carr et al. (2001) proposed one of the first global shape approximation techniques by reconstructing an implicit representation of a shape. The implicit function is defined as

$$f(\mathbf{x}) = \sum_i \alpha_i \varphi(\| \mathbf{x} - \mathbf{c}_i \|_2)$$

where $\mathbf{c}_i$ are the centres of the radial basis functions and α_i the weights for the i^{th}

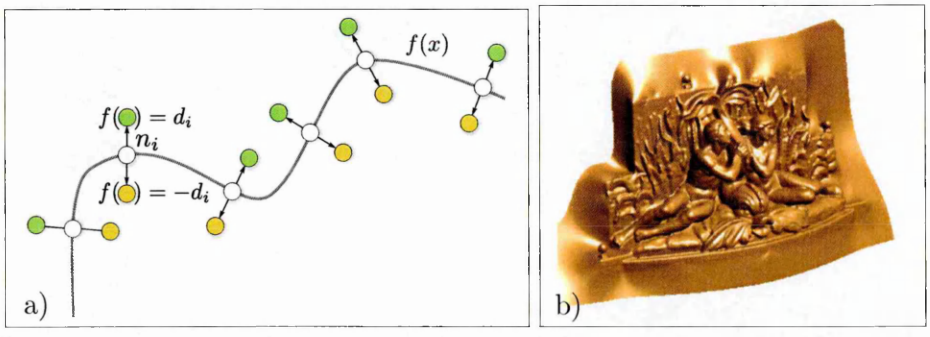


Figure 2.9: a) The implicit function via augmented off-surface points. b) Extrapolation far away from samples. Carr et al. (2001).

RBF center respectively. The approach exploits the included normals for orientation information as shown in Figure 2.9a. The shape function is expected to give zero at the samples $f(\mathbf{p}_i) = 0$ and d_i on the off-surface points $f(\mathbf{p}_i + \epsilon_i \mathbf{n}_i) = d_i$. This enables a convex LSQ minimization task to be defined:

$$\boldsymbol{\alpha} = \arg\min_{\boldsymbol{\alpha}} \sum_i f(\mathbf{p}_i)^2 + (f(\mathbf{p}_i + \epsilon_i \mathbf{n}_i) - d_i)^2 + (f(\mathbf{p}_i - \epsilon_i \mathbf{n}_i) + d_i)^2$$

of which the global solution provides the weightings $\boldsymbol{\alpha}$. The smoothness of the final representation is controlled by the offset distances d_i and the polynomial degree of the RBF. The achieved smoothness and good extrapolation capabilities allow solving sparse sampling issues and also approximating details in high density areas. The distance between the offsets d_i is, however, a critical parameter which leads to artefacts when complex shapes are processed.

Hornung and Kobbelt (2006) proposed a discrete technique addressing the issues with the off-surface distances. The space is partitioned into a *narrow band* voxel grid around the samples and a distance value from its nearest sample is assigned to each voxel (Figure 2.10a). The shape is extracted as the shortest path between the voxels applying

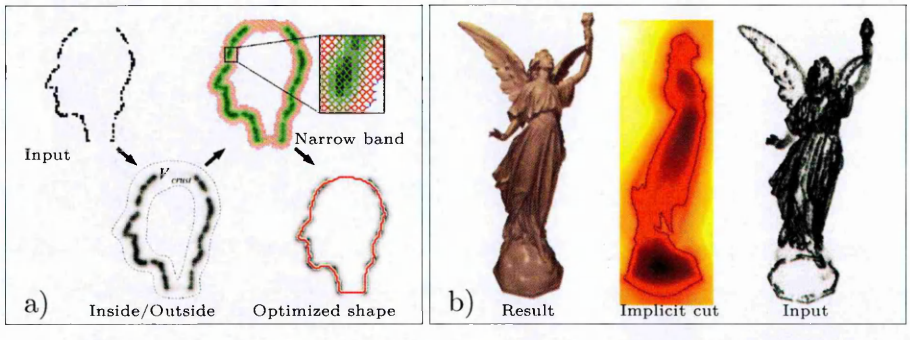


Figure 2.10: a) Narrow band voxel grid around the samples for Graph Cut Hornung and Kobbelt (2006). b) A smooth and global implicit shape extracted via radial basis functions Walder et al. (2006).

the Graph Cut technique from Boykov et al. (1999). Unfortunately, the method suffers from a high computational complexity and is not suitable for datasets with millions of 3D points.

Kazhdan (2005) avoided using off-surface samples by defining the reconstruction task as a partial differential equation (PDE). Fourier basis functions are used to extract the shape which, however, leads to the smooth approximation of edges and corners. Manson et al. (2008) replaced the Fourier bases by wavelets, known for their superior approximation of unit step functions. The difficulty is, however, that the wavelets are separated on each of the three axes (x, y, z) and then combined independently. This leads to good approximation results along the axis alignment but degraded quality for arbitrary-oriented shapes. For these reasons, Manson et al. (2008) also applied post-filtering to smooth the resulting inconsistencies, though this further degrades the quality.

Later, Kazhdan et al. (2006) replaced the Fourier bases by smooth splines while subdividing the space with an adaptive octree. The resulting method is known as the *Poission* (Kazhdan and Hoppe, 2013) reconstruction technique and is widely applied for shape reconstruction.

Walder et al. (2006) proposed a two-step application of RBF where small regions are approximated independently via global RBFs (Figure 2.10b). In the second step, local approximations are combined with compactly supported RBFs. Walder also proposed a novel regression model which forces the gradient of the shape function f to match the sample normals:

$$\nabla f(\mathbf{p}_i) = \mathbf{n}_i.$$

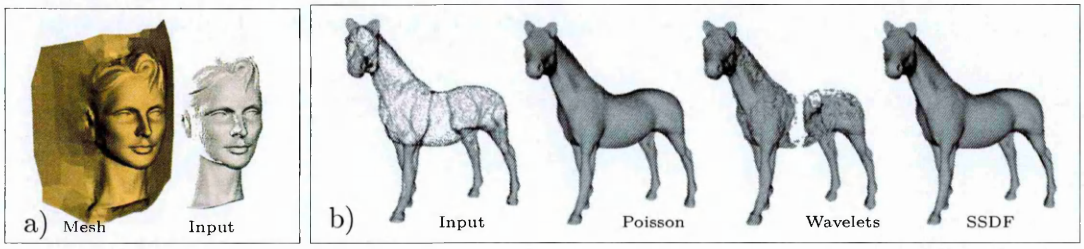


Figure 2.11: a) Reconstruction via SSDF. b) SSDF compared with Poisson and Wavelets Calakli and Taubin (2011).

In the convex optimization task this constraint is integrated as a cost term. A drawback of this method is that the combination of locally defined functions allied with a global smooth RBF leads to over-smoothing. However, Walder's regression model inspired Calakli and Taubin (2011) to apply its discrete form on an octree. Further, Calakli and Taubin (2011) minimized the second derivatives, which led to *Smooth Signed Distance Field* (SSDF) surfaces. Figure 2.11a shows the extrapolation behaviour of SSDF on sparse samples and the accuracy compared to Kazhdan et al. (2006) and Manson et al. (2008) in b. The method is accurate but expects a closed surface, which is also the case for Kazhdan and Hoppe (2013). As a consequence, shape meshes are constructed in non-sampled areas such as doorways and windows (Figure 2.11a).

2.2.4 Piecewise Smoothness Priors

Smoothness priors enable higher shape reconstruction accuracies but often lead to over-smoothing along edges or corners. Human-made environments such as buildings, urban environments or indoor areas often contain planar shapes connected by corners and edges. Boulch et al. (2014) proposed extracting planar clusters from the point cloud and extrapolating the planar arrangements to fill holes. The approach applies combinatorial integer programming (Andres et al., 2012) in order to find a set of possible

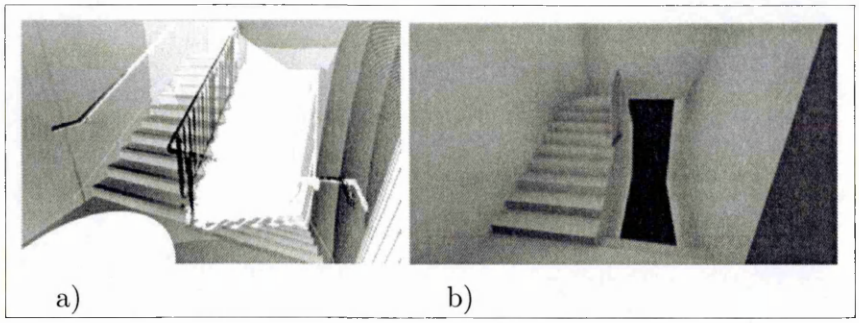


Figure 2.12: Shape reconstruction via combinatorial optimization. a) Input samples, b) reconstruction result Boulch et al. (2014).

intersections in space which leads to good approximations of indoor environments, but finer detail such as staircase railings are ignored as illustrated in Figure 2.12.

A general edge-aware smoothing has been proposed by Oztireli et al. (2009):Robust Implicit MLS - (RIMLS). Applyig RIMLS, iterative reweighing of the sample importance weights is performed giving higher weight to edges. The shape approximation task is similar to the approach of Kolluri (2005) (Figure 2.7b) but is extended by a non-constant weight w_i:

$$f(\mathbf{x})^{t+1} = \frac{\sum_i \mathbf{n}_i \cdot (\mathbf{p}_i - \mathbf{x}) \varphi(\| \mathbf{p}_i - \mathbf{x} \|_2) w_i(\mathbf{n}_i)}{\sum_i \varphi(\| \mathbf{p}_i - \mathbf{x} \|_2)}$$

with $\mathbf{p}_i$, $\mathbf{n}_i$ the i^{th} sample position and normal orientation respectively, and

$$w_i(\mathbf{n}_i) = e^{-\|\nabla f(\mathbf{x})^t - \mathbf{n}_i\|_2^2 / \sigma_{ni}}.$$

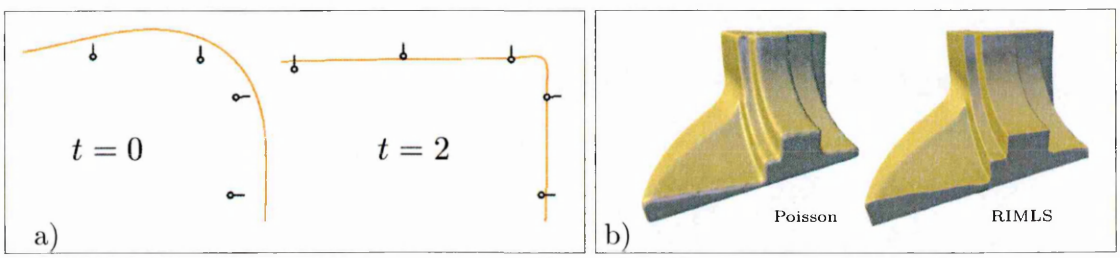


Figure 2.13: a) Edge-aware reconstruction after zero or two reweighting steps. b) Comparison of RIMLS and the Poisson Oztireli et al. (2009).

w_i decreases when the shape orientation deviates from the sample orientation. Figure 2.13a illustrates the iterative behaviour of the reconstructed shape after zero and two refinements (reweighting steps) while Figure 2.13b presents a comparison between the Poisson and RIMLS techniques. A drawback of this method is its sensitivity to irregularities in the sample orientations and because it uses a MLS approach, it is not robust to noisy data.

Avron et al. (2010) addressed the robustness issue by proposing a two-step re-sampling process: i) correcting the normal vectors $\mathbf{n}_i$ and ii) correcting the sample positions $\mathbf{p}_i$ along the estimated normal vectors (Figure 2.14). In the first step a cost function penalizes orientation differences between neighbouring samples. Since only a small number of samples is expected to be located upon an edge or corner, the sparsity-inducing L_1 norm is applied in the cost term. The L_1 based optimization has received a lot of interest in the statistics community (Gonin, 1989; Bach et al., 2011) due to its proven robustness in many application domains. The method is particularly accurate when applied to dense point clouds and outperforms RIMLS as illustrated in Figure 2.15. Unfortunately, the computational overhead is very high which means that only relatively small datasets, i.e. up to around $240,000$ samples, have been used for demonstration purposes requiring a computation time of 22 minutes.

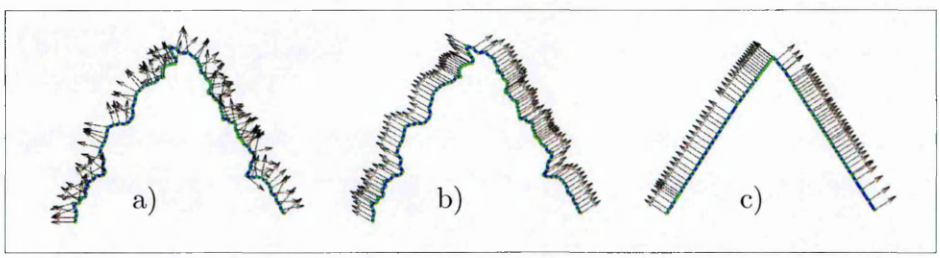


Figure 2.14: A two-step algorithm refining the orientations (b) and the positions (c) of noisy samples (a) in independent processing steps. (Avron et al., 2010).

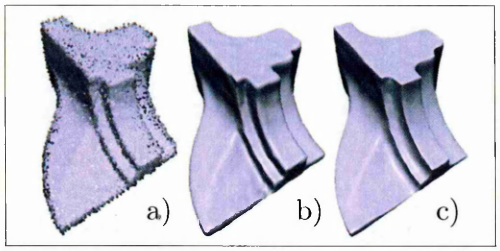


Figure 2.15: a) Noisy data, b) RIMLS outperformed by Avrons method in c) Avron et al. (2010).

Also motivated by the robustness of L_1 cost terms, Bredies et al. (2010) proposed an optimization technique for filtering noisy depth images (Figure 1.5). The technique penalizes the variations in the second derivative of the depth values. Since this will only be non-zero on edges or corners, the sparsity-inducing L_1 norm is chosen. Penalizing variations in the function derivatives is usually referred to as *Total Variation* (TV) minimization in the image processing discipline. However, applying TV to derivatives higher than first order is difficult when noisy data is processed. Bredies et al. (2010) approached this issue by integrating smooth weighting functions onto the samples. They reported very promising results by applying this technique to depth images. In addition, extending this approach to 3D shapes has not been considered so far, which was in part the rationale for the second research objective addressed in this thesis.

2.3 Key Considerations

From this critique of the relevant 3D shape reconstruction literature five critical aspects have been distilled. These are:

1. **Generality** (Gen.): Applicability of the technique to be able to reconstruct both simple and complex shapes.

2. **Computational Speed** (Spd.): Computation efficiency and expandability to parallel computing.

3. **Robustness** (Rob.): Ability to deal with strong noise and/or outliers.

4. **Sparse Measurements** (SM): Ability to reconstruct shapes when only sparse samples are available.

5. **Accuracy** (Acc.): Overall accuracy of the technique.

Table 2.1 summarizes the reviewed methods with respect to these five criteria. The plus

Table 2.1: State-of-the-art shape reconstruction approaches

Technique	1:Gen.	2:Spd.	3:Rob.	4:SM.	5:Acc.
No priors					
α shapes - Edelsbrunner and Mücke (1994)		+++			+
BPA - Bernardini et al. (1999)	+	+++			+
SDF - Newcombe et al. (2011)	+++	++	+		++
Regularity priors					
Clustering - Pauly et al. (2008)			+++	+++	+
Subspace tension - Berner et al. (2011)	+	+	+++	+++	+
Learning clusters - Bao et al. (2013)			+++	+++	+
Local smoothness priors					
MLS - Alexa et al. (2001)	++	+++			++
Point blending - Kolluri (2005)	++	+++			++
APSS - Guennebaud and Gross (2007)	++	+++			++
MPU - Ohtake et al. (2003)	++	+++	+		++
Global smoothness priors					
RBF - Carr et al. (2001)	++	+	+	+	+
Graph Cut - Hornung and Kobbelt (2006)	++		+	+	++
Fourier - Kazhdan (2005)	++	++	+	+	++
Wavelet - Manson et al. (2008)	++	++	+	+	+
Poisson - Kazhdan and Hoppe (2013)	+	++	++	+++	++
SSDF - Calakli and Taubin (2011)	+	++	++	+++	++
Piecewise smoothness priors					
Intersection clusters - Boulch et al. (2014)			+++	+++	++
RIMLS - Oztireli et al. (2009)	++	++			+
L_1 re-sampling - Avron et al. (2010)	++		+++		+++
TV depth approx. - Bredies et al. (2010)		+	+++		+++

signs indicate whether a particular technique is satisfactorily fulfils a certain criterion. This judgement has been distilled mainly from the literature and in certain cases by direct communication with the authors. In particular, the computational speed metric, which is easy to evaluate when the source code for an algorithm is provided. However, it is difficult to appraise from only a publication. In such cases, the computation approach has been analysed by focusing on its capabilities with respect to the computation time. A simple example is when an algorithm applies local smoothness, which scales linearly with the dataset, then its computational speed is shown to have the maximum number of plus (+) signs in Table 2.1.

The overall conclusion is that techniques which do not exploit any priors are inappropriate for either noisy data or sparse samples, with the one exception being the discrete SDF by Newcombe et al. (2011), though this suffers from very high storage requirements.

Regularity priors are stable under the influence of noise or outliers, but are usually designed for very specific scenes such as building facades or vehicles. This restricts the applicability domains.

Smoothness is the most common property observed in any environment, thus integrating smoothness priors enables the accuracy of shape reconstruction to be increased for a large number of applications. While local methods make fast parallel computations possible, they are less suited for noisy or sparse 3D points. Here, global methods perform significantly better. Outstanding techniques are the *Poisson* method from Kazhdan and Hoppe (2013) and the SSDF (Calakli and Taubin, 2011), which are particularly good at reconstructing sparse data with high accuracy but require closed

surfaces, which reduces the generality of the approaches.

Since smoothness priors fail at reconstructing sharp corners and edges, techniques integrating **piecewise smoothness** are in the focus of current research (Strekalovskiy et al., 2014). Clustering samples and searching for intersections of planar segments has been claimed to be robust to outliers but is not applicable to scenes containing fine and coarse details (Boulch et al., 2014). On the other hand, RIMLS from Oztireli et al. (2009) applies iterative edge-aware optimization to reconstruct sharp features, though this technique is not robust to noisy data. In this regard, the highest accuracy is achieved by the L_1 re-sampling approach from Avron et al. (2010) and the TV depth filtering from Bredies et al. (2010), which also applies the L_1 norm. The integration of the TV penalties and L_1 norms is particularly challenging from the computational point of view. In contrast to regularity priors, the smoothness prior is more general and can be applied to a much larger range of environments. It is thus more attractive for 3D shape reconstruction in robotics applications. The planarity regularization via the L_1 norm is a very promising technique to further increase the shape accuracy from general datasets.

2.4 Summary

Several state-of-the-art techniques in shape approximation make use of *a priori* knowledge. For example, detection of regularity patterns in the point clouds allows to extrapolate the shapes despite missing measurements. This approach works well on scenes like a facade with repetitive windows but does not provide benefits on more general environments. For this reason, the smoothness prior, which expects the surfaces

to be smooth, is applied more frequently for indoor and outdoor scenarios. In the context of mobile robotics the integration of smoothness is a promising direction. Therefore, the first research objective focuses on this issue.

Enhancing the smoothness priors to planar surfaces, most evident in indoor and urban environments, has been demonstrated on images up to today. The extension of such planarity-aware, image-based techniques to 3D data is very promising and is therefore addressed by the second research objective. Since the state-of-the-art processes involve optimization strategies which suffer from high computational complexity, the third research question is the reduction of the computational overhead without having to sacrifice the accuracy of the method.

The next chapter will present the methodology established to ensure the correctness and the success of the novel technique and the benchmark strategy when assessing the quality of the 3D shape reconstruction.

3 Methodology

3.1 Framework Structure

The proposed framework in this thesis refers to a structure that processes 3D data to a continuous and discrete surface. The framework is generic in terms of the source of input 3D data, which can be a 3D laser scanner or a camera system. However, the main motivation is to be able to process error-prone data from a stereo camera system, which would enable reliable mobile robots with cameras to be increasingly cost-efficiency. The structure of the framework applied to stereo image data is illustrated in Figure 3.1. Processes **p1** and **p2** are shown in grey as they are not part of the presented framework. Essentially, at least two images (e.g. left and right) are processed to a depth image, where each pixel grey value represents the distance of the 3D point from the camera centre. This allows the computing of a 3D point cloud from either one or more depth images.

The first process *p3*, in the new framework structures the input point cloud into an octree for faster access and more processing efficiency. This process is not critical and is a standard approach in spatial data processing (see Appendix E for more details). Process **p4** represents the core method for 3D shape reconstruction, which contains the three key thesis contributions: 1) implicit surface approximation, 2) integration

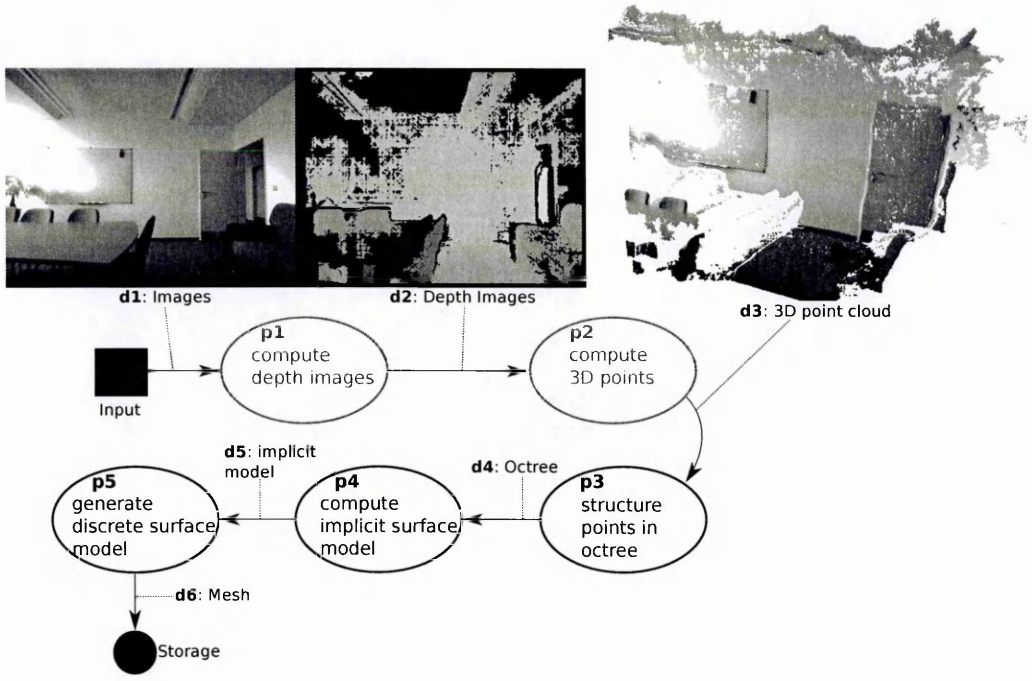


Figure 3.1: The framework applied on stereo image data.

of *a priori* knowledge into the approximation task, 3) efficient numerical optimization technique.

3.2 Integrated Iterative Research

Successful research requires a systematic methodology which helps to keep track of the objectives and to manage the findings. In order to achieve the three research objectives stated in Chapter 1, an iterative methodology is applied in which the literature review, fast idea prototyping, testing and validation are integrated parts of the process. Similar approaches are known from the discipline of project management (Schwaber, 2004). However, the challenge in scientific research is that the impact of a particular idea is difficult to predict. This is addressed by smaller work phases that are carried out iteratively. Figure 3.2 illustrates the herein developed *integrated iterative research* (IIR) steps as one iteration. The IIR methodology relies on the principles of the agile project

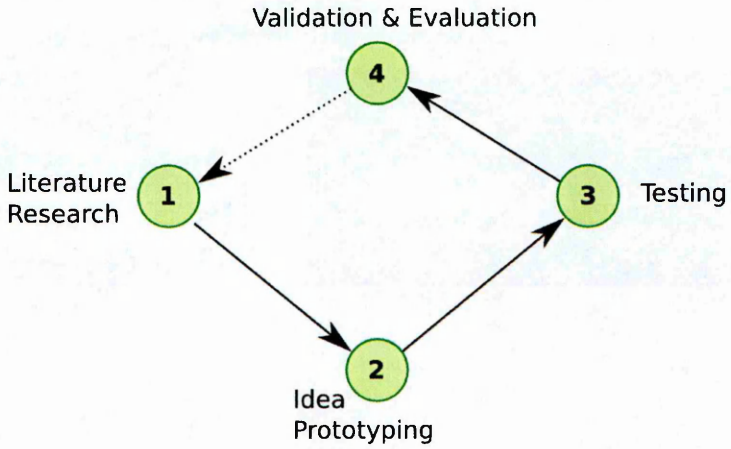


Figure 3.2: Illustration of the IIR methodology.

management, which particularly focuses on experimental software development projects where the outcome is not known *a priori*. A single agile period consists of steps such as i) identifying a task that improves the software, ii) working on the problem for a defined period of time, and iii) reviewing the new functionality and defining the new tasks for the next period. Similarly to this, four research periods have been defined:

1. The first step is the investigation of existing literature regarding the given objective. Since the internet search engines give access to a huge body of research work published as journals, conference proceedings and various web-based resources like blogs, the investigation of a suitable approach for the stated objective is challenging. Section 3.3 presents the established work flow for literature research and introduces the sources and tools applied to manage and to assess new findings. Having identified either a promising approach or strategy which addresses a particular research objective, the first step is to implement a very basic version of the algorithm.

2. The second goal is to assess the suitability of the selected approach and to identify

issues and limitations that are often not mentioned or stressed in the published documents. Since the prototype aims at providing deeper understanding, a programming language with strong visualization support and simple syntax is favoured. The implementation aspects will be discussed in detail in Section 3.4.

3. The third task is to validate the correctness of the prototype implementation. Test scripts, also referred to as *unit-tests*, check the computational correctness of test cases in which the exact results are known *a priori*. By using established test metrics, the performance of the overall framework can then be undertaken. This will be explained further in Section 3.5.

4. The final step is the evaluation of the developed algorithm's implementation and validation, which will be presented in Section 3.6. Synthetic and real datasets are applied in order to reveal the achieved performance of an algorithm. The various 3D test datasets used for the critical qualitative and quantitative analysis of the algorithms are discussed in Sectino 3.6. The ground truth dataset generation process and evaluation metrics are also included.

3.3 Literature Research

Information Sources

Many sources of scientific publications like journal articles, conference proceedings and various web-based resources like blogs are accessible via web search engines. The research community addresses the difficulty to find relevant information by defining *impact factors* (Greenwood, 2007) which help to identify the relevance of authors,

articles, conferences or journals. Nevertheless, characterization of research quality via a single number is critically discussed in the community (Seglen, 1997). In any case, intensive reading, information management and individual critical analysis of the publications is fundamental. Tools such as *Zotero* (Roy Rosenzweig Center, 2015) or *Redmine* (Lang, 2015) help to structure publications and to store individual notes and explanations of the subject. Both tools have been selected since they are web-based and can be accessed from any computer via web browser.

3.4 Idea Prototyping

After identifying a promising approach in the literature, the first step is to implement a basic version for proof-of-concept and to evaluate its suitability. In the computing community Matlab is used as a standard tool for fast prototyping of algorithms. However, in recent years the programming language Python has received a lot of interest. A large set of libraries, simple syntax and excellent support by the web community are the main benefits of Python. In contrast to Matlab, Python and its libraries are free (GPL), the language is object and name-spaces oriented, and allows to create advanced graphical user interfaces (GUI). Programs developed in Python work on Windows, Linux or OS X. Nevertheless, the implementations have been carried out under Windows on a standard PC since this enables efficient debugging and flexible software development. In terms of this research work, Python was used for the initial prototype implementations of the RBF-based approximation as will be discussed in Chapter 4.

3.5 Testing

After a prototype, e.g. approximation via RBF, was developed and validated for its correctness with simple examples, it is then implemented in C++. The main reason for this is the increased computation performance, which exceeds the runtime performance of Python or Matlab.

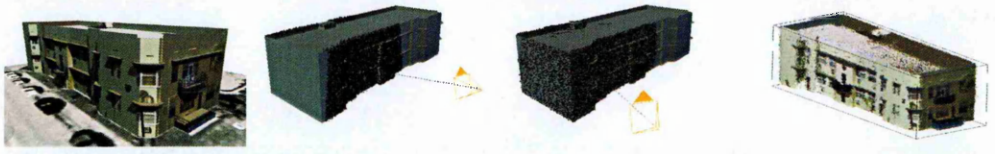
The algorithms implemented in C++ are required to be monitored over time since their stability and correctness have an impact on many parts of the research work. For this reason, *unit-tests* have been adopted. A *unit-test* is a simple test program that performs predefined operations on a software module. In the case of matrix-vector multiplication, a known vector $\mathbf{v}$ and matrix A are multiplied:

$$A\mathbf{v} = \mathbf{r}$$

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \cdot \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} a_{11}v_1 + a_{12}v_2 \\ a_{21}v_1 + a_{22}v_2 \end{pmatrix}.$$

Since the exact outcome is known, the multiplication results are automatically proven to be correct by the *unit-test* program. When one of the elements in $\mathbf{r}$ does not correspond to the expected value, the test-program exits with an error that generates an email alert for the developer.

The *unit-test* approach validates a particular program's correctness but does not investigate the performance of the full algorithm. In the case of RBF-based approximation, for example, in which measurements from a known model are perturbed by noise, evaluating its correctness falls within the scope of the evaluation step. This will



a) b) c) d)

Figure 3.3: Virtual cameras (b-c) scan a 3D model (a). d) The simulated 3D point cloud.

be discussed in the next section.

3.6 Validation and Evaluation

In order to make sure that the implementations are in fact correct, *unit-tests* and evaluation on synthetic and real datasets are combined. While the *unit-tests* ensure that low-level computations such as optimization solvers are correct, the data-based evaluation proves that the developed shape reconstruction algorithm is correct and competitive.

3.6.1 Evaluation Framework

After the shape approximation framework was established, its accuracy and computation times, which are standard performance metrics for shape reconstruction (Strecha et al., 2008), are evaluated. This step naturally involves the comparison of the reconstruction result with *a priori* known results, the ground truth. Unfortunately, ground truths for large scale 3D reconstruction are generally not available in the community. Several research groups have published datasets acquired by laser scanner devices or the Kinect (Microsoft Inc., 2015), but they only provide ground truth data for the pose of the moving sensor (Sturm et al., 2012).

To perform quantitative evaluation, a virtual range camera has been implemented

simulating temporal data acquisition during a movement (Figure 3.3). In the first step, a 3D model from the 3D warehouse (Google Inc., 2014) (Figure 3.3a) is sampled by a range camera from different locations (Figure 3.3b-c). Each image is perturbed with Gaussian noise and the 3D points are aligned in a common coordinate frame (Figure 3.3d). The implementation of the virtual camera strategy has been performed in the rendering engine Blender (Blender Online Community, 2016). Blender is an open-source project delivering high-quality ray tracing rendering for low-cost film productions. It is extendible with custom scripts that enable it to access the properties of the camera and the scene and thus to simulate range camera measurements and the scanning process. By applying this approach, it is possible to load a common mesh-based 3D model into Blender and to define the trajectory of the camera movement during the animation. When the animation is started, a 3D point cloud is generated using the camera position at each frame. Finally, a set of 3D point clouds is available, one for each time point respectively. The full assessment framework is summarized in Figure 3.4 and has four underlying processes $p1$-$p4$ which perform the following tasks:

- **p1**: Given a 3D model, the polygon-based shape is re-sampled with at least 10^6 points. The resulting point cloud, labelled as **d2** in Figure 3.4, is defined to be the ground truth and is passed to task **p4** for the assessment of the reconstruction result.

- **p2**: The 3D model is scanned by the synthetic camera and stored as a point cloud in a file.

- **p3**: This is the main shape reconstruction process. It uses the point cloud from

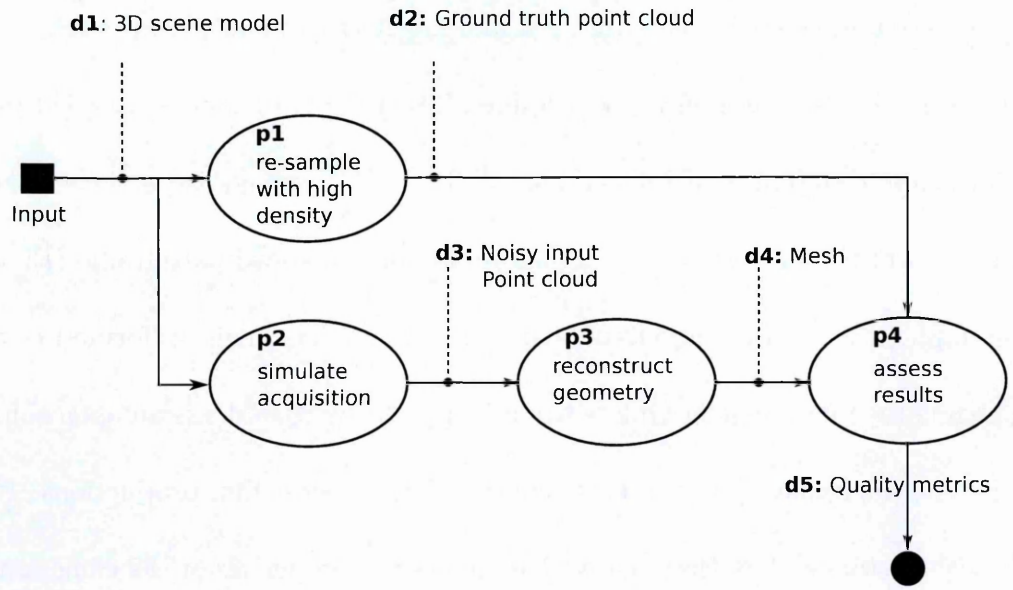


Figure 3.4: The process overview developed to assess the shape reconstruction quality. **p** represents a process task and **d** is the data type which is transferred from one process to another.

the process **p2**, approximates the shape and generates a mesh which is further evaluated by **p4**.

- **p4**: The resulting mesh is re-sampled in the same way as in **p1**. Point-to-point error distribution with respect to the ground truth from **p2** is calculated.

3.6.2 Performance Metrics

The last step **p4** computes the performance metrics representing the success of the reconstruction process **p3**. Besides the perceptual inspection of the reconstruction shape results, numerical evaluation using appropriate performance metrics is undertaken to enable more meaningful information to be extracted and to assess the success of the new shape reconstruction process. Similar to Berger et al. (2013), the Euclidean distance between the reconstructed surface and the ground truth surface is computed. For each sample from **d4** its next neighbour in **d2** is found. Furthermore, the statistical error

distribution including cumulative statistics is extracted. A similar approach has been applied by Strecha et al. (2008) for the evaluation of multi-view stereo algorithms. In summary, the following performance metrics are applied:

- **Visual inspection**: Shape reconstruction results are visualized using OpenGL. Viewing the shapes in detail allows to quickly detect inconsistencies in the reconstruction process.

- **Spatial deviation**: Spatial deviation of the reconstructed shape from the original model. This is the main metric investigated in **p4**. The nearest neighbour from the ground truth point cloud is selected for each mesh vertex. The spatial error distance is shown in terms of colour intensity in Figure 3.5a. The model is visualized in a different colour where the intensity of red represents the deviation from the ground truth.

  Using common statistical visualization standard tools (Tufte, 1990) such as median and variance plots (Figure 3.5b), error histograms (Figure 3.5c) and the cumulative distribution plots (Figure 3.5d) are generated. These provide additional information about the intensity of the reconstruction errors and also their spatial distribution.

- **Median and variance**: After computing the spatial deviations between **d4** and **d2**, the error distribution information is summarized by the median value. First, all the N distances are sorted in ascending order. Second, the central sample at the list position $N/2$ is taken as the Median Distance Error (MDE). The benefit of MDE is its robustness to large outliers in the spatial distances. In cases

where large reconstruction errors occur at only one position, the corresponding MDE metric is less likely to increase than the mean value. Additionally, MDE is augmented by the error variance in Figure 3.5b, in which four exemplary methods are shown for comparison.

- **Distribution visualization**: In statistics, the probability density function plays an important role when evaluating the nature of a random process. Its equivalent discrete representation is the histogram, shown in Figure 3.5c. The histogram reveals how well the reconstructed shape aligns to the ground truth surface. The peak illustrates the most frequent error made by the reconstruction process. However, intuitive comparison using the histograms is difficult. This is addressed by the cumulative distribution which can directly be computed from the histogram. In the cumulative distribution graph in Figure 3.5d a superior method would rise sharply to 100% and a method of low quality would be less steep and thus reach 100% fraction at a larger error value. In particular, the vertical axis expresses the percentage of samples with a smaller error than the corresponding value on the horizontal axis. In summary, the higher the accuracy, the steeper is the cumulative curve in Figure 3.5d and the smaller the median value in Figure 3.5b.

- **Runtime**: The time an algorithm requires to accomplish the computation is measured in milliseconds using the CPU clock. This empirical metric enables the evaluation of the achieved computational complexity. For all experiments the computing platform was an Intel i5 CPU with 4 GB of memory. Applying platforms with higher performance will accordingly improve the runtime performance metric.

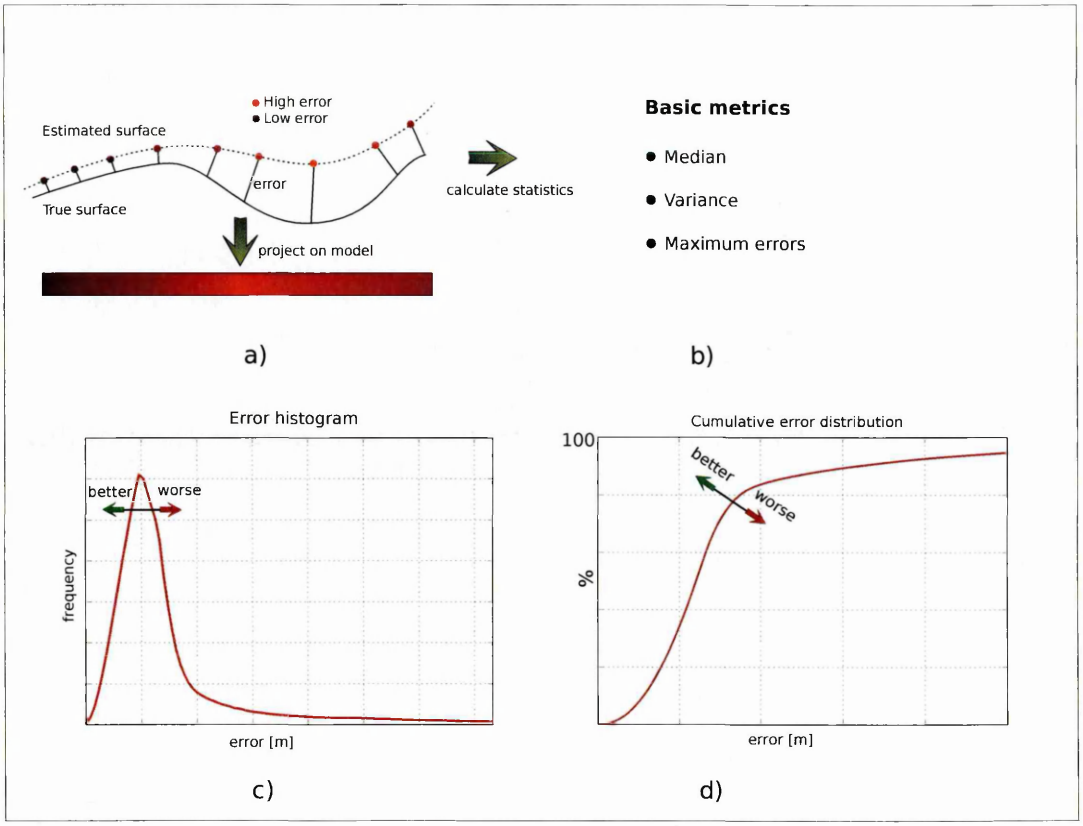


Figure 3.5: a) Calculating error distances between two shapes and projecting the colour on the model (bar). b) Median, variance and the maximum error. c) Error histogram. d) Cumulative error distribution.

3.7 Datasets

3.7.1 Synthetic 1D Signal

The aim of this research is to perform reliable shape approximation from 3D samples. Since the new framework relies on foundations from signal recovery theory, it is beneficial to critically evaluate its performance initially on a 1D sinusoid signal. This simplifies the visualization of the approximation results as a standard plot and enables qualitative validation of the correctness of the process. Thus, a synthetic 1D signal is used in Chapter 4, where the basic properties and drawbacks of signal reconstruction using RBF are discussed. Figure 3.6 illustrates samples from a sinusoid signal, which have

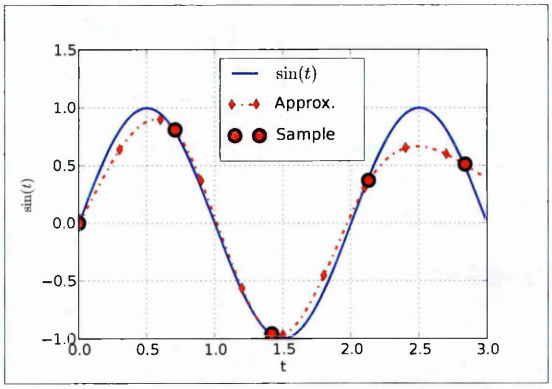
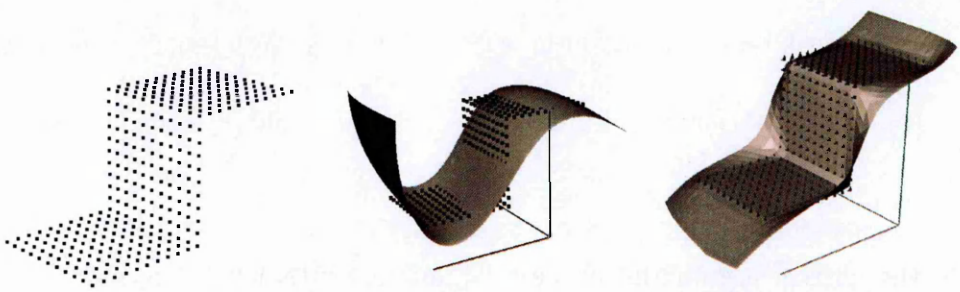


Figure 3.6: A simple synthetic 1D signal as a verification dataset for RBF-based approximation.

been used for the approximation.

3.7.2 Synthetic 3D Step Function

To validate the correctness of the implicit shape reconstruction from the 3D data, samples from a simple step function are used as a validating dataset (see Figure 3.7). The benefit of this approach is that the dataset size and noise can easily be controlled. The model also contains planar areas and sharp corners, so the extrapolation away from the samples can thus be evaluated. This dataset has been used in Chapter 5 to evaluate different kernels and regularization techniques for 3D shape approximation.



a) b) c)

Figure 3.7: a) Input 3D samples, b) a poor shape approximation, c) a good shape approximation.

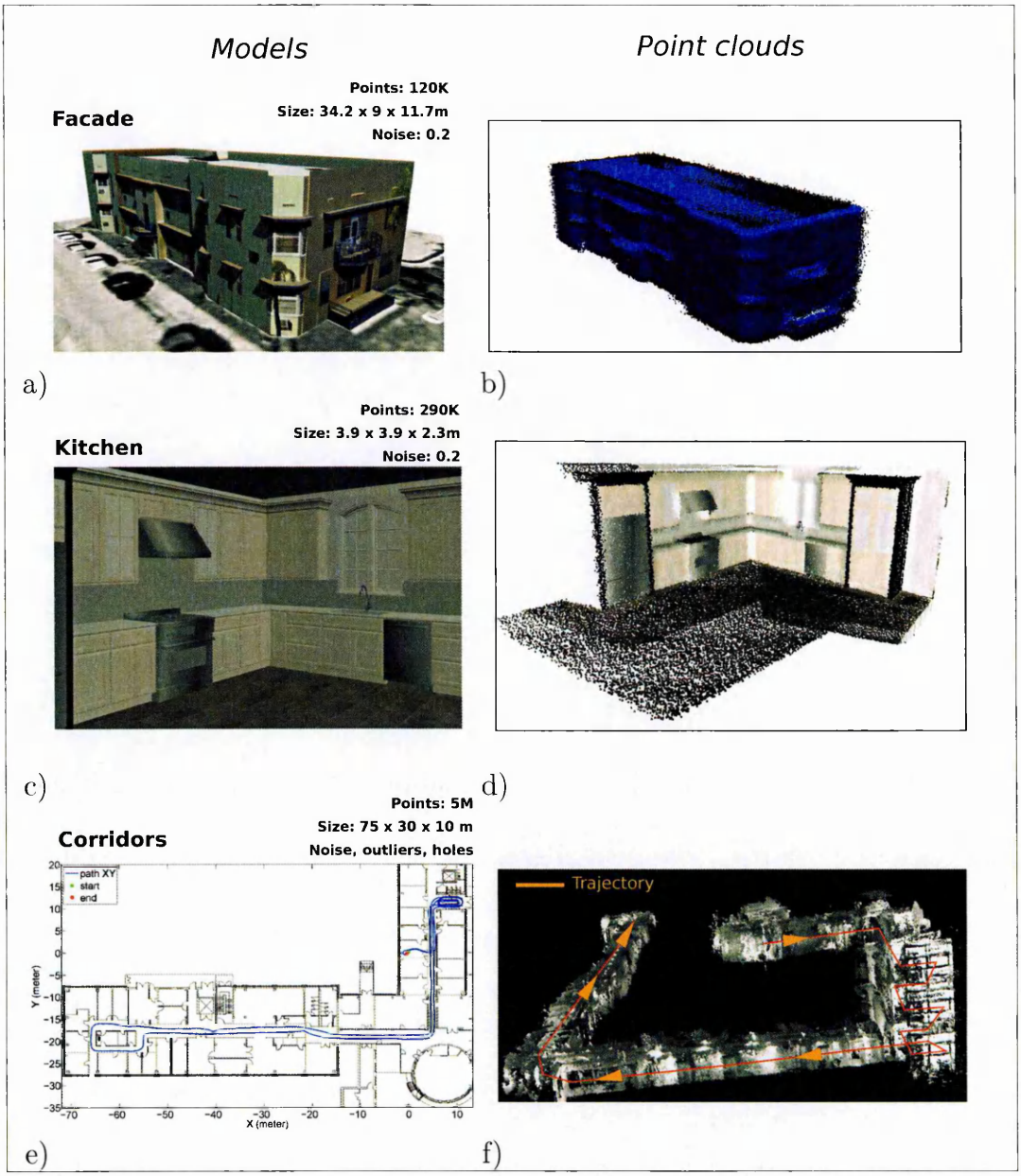


Figure 3.8: Synthetic datasets (a-d) and datasets obtained from stereo (e, f).

3.7.3 Complex 3D Environments

The final evaluation phase for the developed TVL_1 knowledge integration framework was performed on two synthetic and one physically acquired point cloud datasets. A major limitation with currently available 3D datasets in the community, is that no ground truth is available which means, that even if some precise 3D samples are

provided, the true position and orientation of the sampled shape is unknown. This however, is required in order to perform rigorous quantitative evaluation of the shape approximation algorithm. Thus, two synthetic point clouds have been simulated using a ray tracer, which is referred to as a virtual range camera. The simulated scanning process is described in Section 3.6. The scenes are firstly a facade of a building, which contains planar and sharp corners, which is generally the most difficult to reconstruct in the shape approximation process. The second scene is a simulated indoor scan of a kitchen, which also contains challenging planar and sharp edges, together with small objects.

These two datasets have been specifically designed to reflect a broad range of requirements often imposed on 3D shape reconstruction. Thus, when an algorithm is performing well on both these datasets, it is a pragmatic generalisation to claim they are also expected to perform well on real data obtained from stereo cameras.

The third dataset has been acquired from a mobile stereo camera system fusing measurements from an inertial measurement unit (IMU). Each stereo image pair has been processed by the *semi global matching* (SGM) algorithm (Hirschmüller, 2011) to 3D points in real time. The alignment of the point clouds in the global coordinate frame has been achieved by the sensor fusion technique of the *integrated positioning system* (IPS) (Griessbach et al., 2014). IPS provides a high-accuracy trajectory of the camera movement (Figure 3.8e) which enables the alignment of the 3D point clouds from each image frame into a global consistent coordinate frame. Using the synthetic simulation and the IPS, three scenes have been developed for benchmarking purposes that represent indoor and outdoor urban environments. These are:

- *Facade*: A mesh model (Figure 3.8a) from Google's 3D warehouse (Google Inc., 2014) is scanned by a virtual range camera in the artistic 3D modelling software Blender (Blender Online Community, 2016). During the scanning process the measurements of the building with dimensions of $9 \times 5 \times 6.5m$ are perturbed by normally distributed noise with $\sigma = 0.4m$. Sharp corners and edges make this point cloud (Figure 3.8b) challenging to be approximated without over-smoothing either the edges or the corners.

- *Kitchen:* Also obtained from the 3D warehouse, the kitchen model is shown in Figure 3.8c and 3.8d. The object with the dimensions of $4 \times 4 \times 2.6m$ has been scanned by a virtual camera, and a noise of $\sigma = 0.05$ cm has been added, which is expected from real sensors when indoor close-range environments are scanned.

- *Corridors:* IPS and SGM have been used to acquire this dataset. The resulting point cloud (Figure 3.8e) suffers from strong noise and many outliers caused by reflections, varying illumination and fast camera movements. The sampling density varies strongly since due to fast movements and motion blur several areas have not been reconstructed by SGM. Difficult lighting conditions and occlusions further lead to holes or even the absence of complete walls. Figure 3.9 shows the particular areas selected for perceptual assessment from this dataset. These are a large entry into a hall (1), transparent windows from the corridor into the hall (2), transparent glass door (3) and a corridor with high lighting variations (4).

The presented datasets and the evaluation methodology have been applied to the developed framework and the results will be presented in Chapter 7.

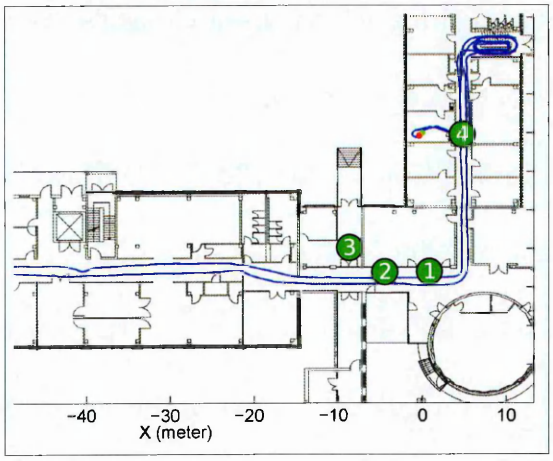


Figure 3.9: Selected locations from the corridor dataset. 1: Large hall entry, 2: Windows from corridor into the hall, 3: Glass door, 4: Corridor.

3.8 Summary

The developed research methodology presented in this chapter divides the complex research task into four distinct parts: literature study, prototype development, testing for correctness and performance evaluation. For each of them a set of tools has been developed which facilitates continuous improvements. The following chapter will lay out the theoretical concepts for RBF-based data approximation, which is fundamental for understanding the new shape approximation algorithm.

4 Fundamentals in RBF-Based Approximation

In Chapter 2 the need for the integration of *a priori* models (priors) in the shape reconstruction technique has been justified. It is stated that the exploitation of smoothness priors is the most versatile model for indoor and outdoor environments and is thus of particular interest for mobile robotics. Smoothness priors can be computed in a robust way using RBF, though over-smoothing is a major consideration since this tends to *smooth out* information and must be avoided. As discussed in Chapter 2, regularization methods are a promising approach to avoid over-smoothing and to preserve sharp edges.

This chapter introduces the application of RBF to reconstruct simple one-dimensional (1D) data and it demonstrates how to extend the optimization technique with non-smooth regularization. One-dimensional data is selected for the introduction since the experiments enable a clear visual analysis of the reconstruction process. The discussed behaviour of the RBF-based reconstruction techniques does not change when the dimensionality of the data is increased.

Regularization techniques have been introduced in order to enable a solution for ill-posed problems (Wahba, 1990). These occur when the number of measurements is lower than the number of unknowns and some *a priori* knowledge must be used to obtain a valid solution (Bickel, 2007). In shape reconstruction regularization enables continuous surfaces to be reconstructed from scattered and non-uniformly distributed 3D points.

As will be shown, the regularization helps to increase not only the reconstruction quality but also the stability of the process. The stability of the RBF reconstruction process is an important issue. If neighbouring samples are too close to each other, the system interpolates redundant measurements and affects the condition of the system negatively.

4.1 Radial Basis Functions

RBF interpolation has been initially studied by Hardy (1971), Duchon (1977) and Pratt (1987). Working in the field of cartography, Hardy empirically discovered that a linear combination of multi-quadrics, now known as multi-quadric RBFs, produces an interpolant that fits all the scattered point data. Duchon (1977) and later Madych and Nelson (1990) studied the globally smooth class of RBFs proposing the globally defined *Thin-Plate splines* (TPS) for scattered data approximation and numerical solutions of coupled partial differential, stochastic equations. However, globally defined RBFs are not feasible for large datasets, so Wendland (1995) introduced the RBF with compact support, which has received considerable attention. Later, Wendland (2004) studied the stability issues related to RBF interpolation and derived formal stability criteria using the Fourier-Bessel transform. This enables the characterization of the regression system with only the expected sampling density given, so no numerical experiments are necessary. The underlying research work is inspired by these advances in the RBF approximation and focuses on three RBF types in particular:

1. State-of-the-art Gaussian RBF,

2. *Thin-Plate splines* with global smoothness properties and

3. Wendland's RBFs, which enable efficient regression and the control of the surface smoothness.

More details on these aspects will be provided in the following sections.

4.1.1 Interpolation with Radial Basis Functions

The general goal is to find a function $f : \mathbb{R}^d \mapsto \mathbb{R}$ from a set of N sample values $y_i\mathbb{R}$ at $x_i \in \mathbb{R}^d$. The core idea of RBF-based interpolation is that the function $f(x)$ is represented by a linear combination of M weighted basis functions (Takens, 1968):

$$f(x) = \sum_{j}^{M} \varphi(x, x_j)\alpha_j. \tag{4.1}$$

Each of the basis functions $\varphi(x, x_j)$ is centred at x_j and essentially computes the similarity between x and x_j. One possible form for φ is $\varphi(x, x_j) = e^{-\|x-x_j\|_2/s}$, which is similar to the Gaussian distribution with $s = \sigma$ influencing the width of the support. Consider a set of x_j on a circle around x, then $\varphi(x, x_j)$ will give the same value for all x_j. This is why the Gaussian as well as other RBFs considered in this chapter are called radial basis functions.

The underlying interpolation concept of RBF is illustrated in the one-dimensional example in Figure 4.1, in which f is defined as the sum of all given Gaussians with their respective weights α_j. In this work it is assumed that the widths of the basis functions are known *a priori*, so only the weighting factors α_j are to be determined leading to $f(x)$. The task is therefore to perform regression over N samples and to find

M weights via the minimization of

$$\min_{\boldsymbol{\alpha}} \sum_{i}^{N} (y_i - \sum_{j}^{M} \alpha_j \varphi(x_i, x_j))^2$$

where y_i is the i^{th} measured sample at position x_i. This can also be rewritten in matrix-vector form as

$$\min_{\boldsymbol{\alpha}} \| \mathbf{y} - K\boldsymbol{\alpha} \|_2^2 \tag{4.2}$$

where K is often referred to as the design or *kernel* matrix with $K_{i,j} = \varphi(x_i, x_j)$. The solution to (4.2) is

$$\boldsymbol{\alpha} = A^{-1} K^T \mathbf{y} \tag{4.3}$$

with $A = (K^T K)$, which is the well-known linear *least squares regression* (Hastie et al., 2008). An important aspect of this technique is that $f(x)$ is not restricted to being linear. This linear representation of non-linear functions will form the basis of the

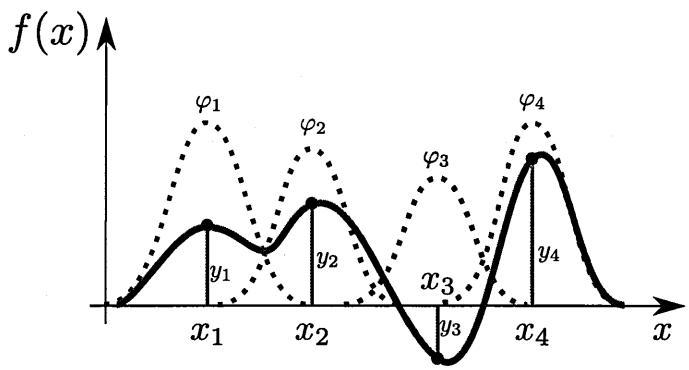


Figure 4.1: Function f constructed by a weighted linear combination of Gaussian radial basis functions.

subsequent 3D shape approximation process. This mathematical framework is able to handle datasets of arbitrary dimensions that only affect the type of the basis functions. Since the number of unknowns is only connected to the number of samples, the technique is very flexible and will be used in successive chapters to approximate surfaces from 3D samples. By applying this approach, the choice of suitable basis functions remains an open question which will be now discussed.

4.1.2 Selected Radial Basis Functions for Geometric Approximation

In the last two decades several types of RBFs have been proposed for different applications, the most popular being Gaussian RBF, which is very commonly used in the machine learning community. Three RBF types have been selected for shape approximation applications:

1. Gaussian since it is the state of the art,

2. *Thin-Plate* splines because of their global smooth properties and

3. compactly supported RBFs which enable sparse regression systems to be created and the smoothness to be controlled.

RBFs are in principle derived by minimizing specific function norms. In the case of the *Thin-Plate*, the global smoothness as well as the invariance to rotations and translations are the key features to be exploited. Similarly, Wendland (1995) proposed RBFs with compact support where the compact non-zero domain of the basis function is formed by a polynomial of a predefined degree. As will be shown in Section 4.1.3, lower degree polynomials increase the stability of the interpolation but they also restrict the

number of the derivatives on $f(x)$, which reduces the smoothness of the reconstructed shape.

Radial Basis Functions with Compact Support

Consider the Gaussian approximation at x_1 in Figure 4.1, where the thick line represents the function response, which is a weighted sum of RBFs φ_i. The function response, namely the thick line, is defined as $f(x_1) = \sum_j^M \varphi(x_1, x_j)\alpha_j$. However, because of the restricted support, $\varphi(x_1, x_j)$ gives a value close to zero for $j = 3$ and $j = 4$. If very small values from φ are neglected, the kernel matrix values $K_{1,3} = \varphi(x_1, x_3)$ and $K_{1,4} = \varphi(x_1, x_4)$ are zero and are not required to be evaluated when computing $f(x_1)$ or performing the regression. Research on sparse linear algebra enables very large matrices to be processed by exploiting the sparsity of K (Heroux et al., 2005). Unfortunately, the Gaussian can only be truncated far away from the centre, which significantly increases the support. This is why Wendland's compactly supported RBFs (CSRBFs) are popular in physical computing since they provide sharp support boundaries.

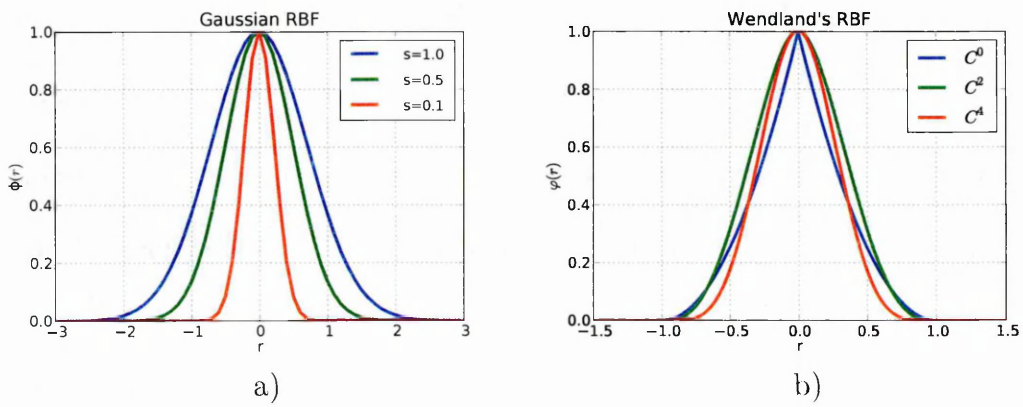
An additional requirement for CSRBF is that the kernel matrix K remains positive definite in any dimension d of $x \in \mathbb{R}^d$, which is the case for Gaussian RBF. This is one of the reasons why Gaussian RBF is widely applied. In contrast, the positive definiteness cannot be guaranteed in all dimensions for the CSRBF (Wendland, 2004) which means specific kernel types need to be constructed in each dimension. Table 4.1 lists selected CSRBF functions (Wendland, 2004) where $\doteq$ denotes equality up to a positive constant factor. The $()_+$ operator is the positive definiteness constraint and can be interpreted as $(x)_+ = \max(0, x)$. Several CSRBF types are presented for the dimensions $d = 1$ and

Table 4.1: Compactly supported functions of minimal degree

Space dimension	Function	Smoothness
$d = 1$	$\varphi(r) \doteq (1-r)_+$	C^0
	$\varphi(r) \doteq (1-r)_+^3 (3r+1)$	C^2
	$\varphi(r) \doteq (1-r)_+^5 (8r^2+5r+1)$	C^4
$d \leq 3$	$\varphi(r) \doteq (1-r)_+^2$	C^0
	$\varphi(r) \doteq (1-r)_+^4 (4r+1)$	C^2
	$\varphi(r) \doteq (1-r)_+^6 (35r^2+18r+1)$	C^4
	$\varphi(r) \doteq (1-r)_+^8 (32r^3+25r^2+8r+1)$	C^6

$d \leq 3$. $d = 3$ is of special relevance for 3D shape approximation. The continuity C^m or the polynomial degree m affect the type of the RBF, which is addressed by different rows in the table. More practically, the higher the polynomial degree of a RBF, the more complex shape variations that can be approximated. The RBF must be of at least $m = 2$ in order to calculate the second derivative of the reconstructed function.

Figure 4.2 illustrates the Gaussian and the compactly supported RBF for different C-continuity and scaling (support-width). Similar to the case of Gaussian RBF, the scaling may be applied to the compact RBF by normalizing the argument $r_s = \frac{r}{s}$, so a larger s makes the curves broader while a smaller value narrows their width.



a) b)

Figure 4.2: a) Gaussian RBF with different scales s. b) Compactly supported RBF with different continuity properties.

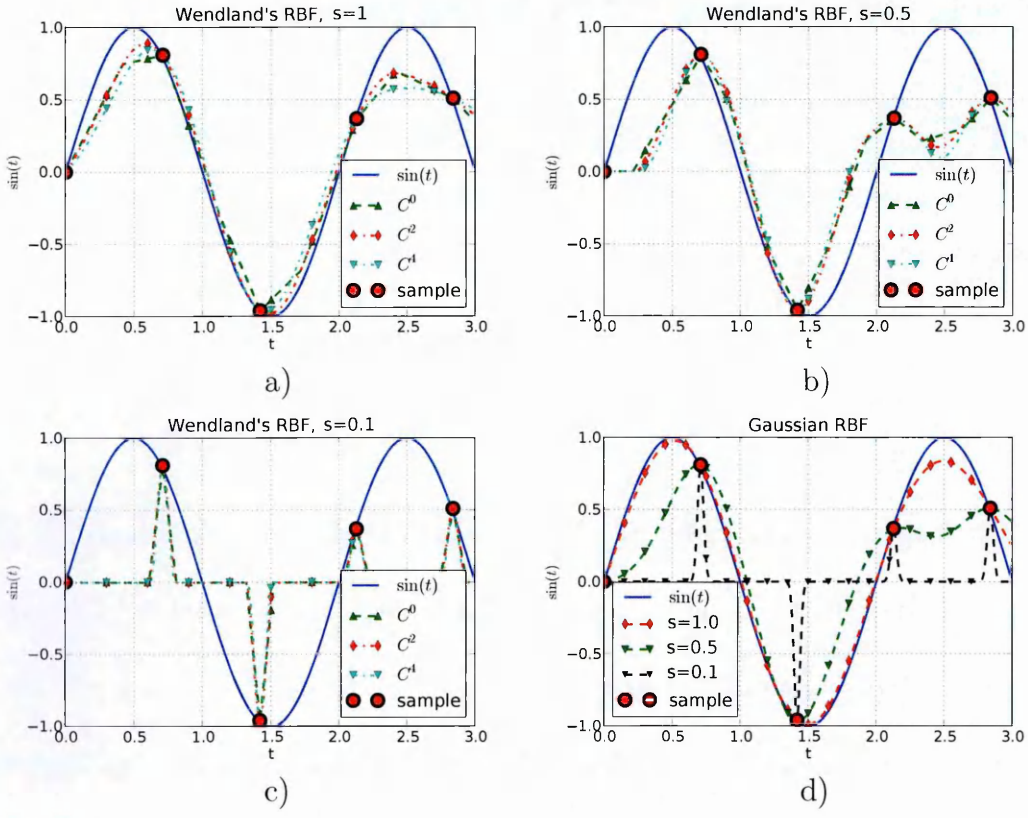


Figure 4.3: a-c) Compactly supported RBFs with different scales and continuities. d) Gaussian RBF with different scales s.

An important restriction of compactly supported RBFs is that the approximation quality is sensitive to proper scaling of the basis functions. Figure 4.3 illustrates the behaviour of the Gaussian and Wendland's RBF with different scaling parameters applied to interpolate a 1D sinusoidal signal. Sin(t) signal has been selected for visualization purposes since the visualization of the approximation quality is intuitive. These properties do not change when data of a higher dimension, eg. $\mathbb{R}^3$, is processed, though higher dimensions make the visualization much more challenging and as a consequence will not be considered further in this chapter.

From the scale variation results in Figure 4.3 it can be concluded that a small scale s leads to poor interpolation between the samples while higher s values increase the

smoothness of the interpolant. If, however, s becomes larger than the distance between two RBF centres, the regression will become unstable. This leads to rank reduction in the kernel matrix K in (4.3) and strong amplitudes in the interpolated function are more likely. A more detailed discussion concerning the instability issue and selection of the best scaling parameters will be given in Sections 4.1.3 and 4.2.

Thin-Plate RBF

Initially introduced by Duchon (1977), *Thin-Plate splines* have been used for solving heat or more general differential equations in well-proven applications like meteorology (Wahba and Wendelberger, 1980). The aim was to find a basis function with maximal smoothness with respect to a specific derivative degree m. In the case of $m = 2$ and for two dimensional data $d = 2$ the minimized cost function is

$$J_{m=2}^{d=2}(f) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(\frac{\partial^2 f}{\partial x_1^2} + 2\frac{\partial^2 f}{\partial x_1 x_2} + \frac{\partial^2 f}{\partial x_2^2})dx_1 dx_2. \tag{4.4}$$

Duchon (1977) showed that a minimization problem of the type

$$\sum_{i=1}^{n}(y_i - f(x_{1i}, \cdots x_{di}))^2 + \lambda J_m^d(f) \tag{4.5}$$

with λ as a weighting parameter of the penalty J has a unique minimizer f_λ with the representation

$$f_\lambda(x) = \lambda\sum_{i=1}^{N}\varphi(x, x_i)\alpha_i + \mathbf{c}^T\mathbf{p}(x) \tag{4.6}$$

with $\mathbf{c}^T\mathbf{p}(x)$ as a polynomial with M_p variables.

The proposed basis functions $\varphi(r)$ with $r = \| x - x_i \|_2$ are

$$\varphi(r) = \begin{cases} r^{2m-d} \log(r) & \text{if } d \text{ is even} \\ r^{2m-d} & \text{otherwise.} \end{cases} \tag{4.7}$$

As shown in Figure 4.4, unlike the Gaussian RBF the *Thin-Plate* basis function attains zero at $r = 0$. This leads to a non-positive definite basis function matrix K in (4.2) which cannot be solved. Wahba (1990) showed that this limitation can be resolved by adding a polynomial to f, which is $\mathbf{c}^{\mathbf{T}}\mathbf{p}(\mathbf{x})$ in (4.6). One constraint to this solution is that the polynomial coefficients must be orthogonal to the $\boldsymbol{\alpha}$ coefficients. The reason lies in the foundation of the *reproducing kernel Hilbert space* theorem (Wahba, 1990). The function of interest f is defined as

$$f(x) = \langle \varphi(x), \alpha \rangle, \tag{4.8}$$

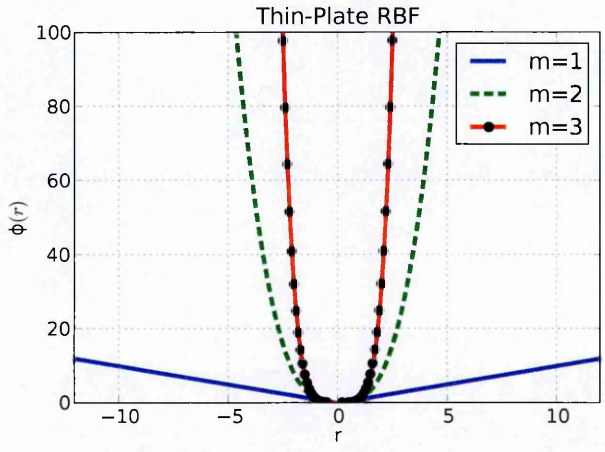


Figure 4.4: The *Thin-Plate* function value $\varphi(r)$ with different m values defining the C^m continuity.

where $\langle \cdot, \cdot \rangle$ is formally the inner product and is practically stated as a finite sum over all elements α_i in (4.1). Including the additional polynomial leads to

$$f(x) = \langle \varphi(x) + p(x)\mathbf{c}, \alpha \rangle, \tag{4.9}$$

which does not hold and gives a different representation of f. So by restricting the polynomial coefficients $p(x)$ to be orthogonal to α, namely $\langle p(x)\mathbf{c}, \alpha \rangle = \mathbf{0}$, the inner product in (4.8) is upheld. Applying this to practical linear algebra computations, the orthogonal polynomials are constructed by introducing a constraint into (4.2):

$$\sum_{i}^{N} \sum_{q}^{M} c_q \alpha_i = 0 \tag{4.10}$$

resulting in a matrix representation

$$\begin{pmatrix} K & B \\ B^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{\alpha} \\ \mathbf{c} \end{pmatrix} = \begin{pmatrix} \mathbf{n} \\ \mathbf{0} \end{pmatrix}, \text{ with } B = \begin{pmatrix} \mathbf{p(x_1)^T} \\ \dots \\ \mathbf{p(x_N)^T} \end{pmatrix}. \tag{4.11}$$

The degree of the polynomial in $\mathbf{c}$ cannot be higher than $m - 1$ as the data can only be fitted if the underlying sample manifold is less than m.

Figure 4.5 illustrates the behaviour of the *Thin-Plate* for different values of m. While the C^1 interpolant directly connects the samples, the C^2 curve is everywhere smooth and is apparently a better choice for reconstructing smooth signals like $sin(t)$ from a relatively small number of samples.

The main drawback of the *Thin-Plate* RBF is its global support. Since the basis

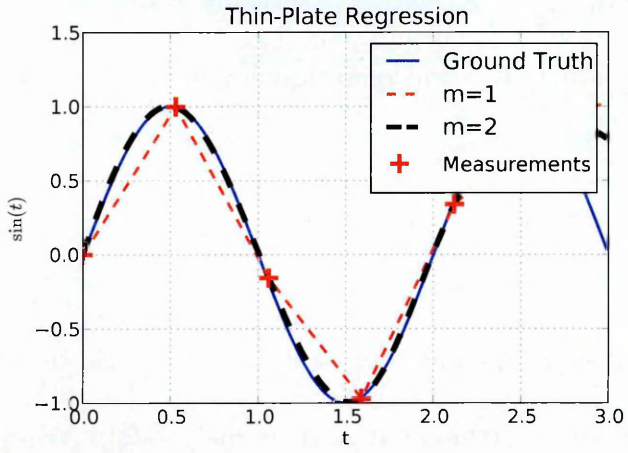


Figure 4.5: The *Thin-Plate* regression of $\sin(t)$ from sparse samples.

functions are non-compact, even a small change in a single centre affects the entire interpolation result. Further, K can never be sparse, so the application of the *Thin-Plate* kernel to large datasets is difficult. However, the focus of this thesis is on the efficient processing of large 3D point clouds, so applying global RBF is impractical.

While exhibiting both good shape approximation performance and stability, *Thin-Plate* RBF has efficiency limitations which leads to the question of how best to apply compactly-supported RBF without compromising these attractive properties. These issues will be discussed and critically evaluated in the next section.

4.1.3 Stability and Error Bounds

Stability

When applying RBF for shape approximation, it is important to be aware of instabilities, which may occur if the data samples are redundant or the sample-to-sample distance is much smaller than the RBF support parameter s. To provide an insight, an example that demonstrates instabilities on the interpolation of a $\sin(t)$ curve is considered. Since

the RBF interpolation concept does not depend on the dimension of the data, the simple one-dimensional sinusoid signal is well-suited to illustrate the benefits and limitations of a RBF valid for all dimensions. In contrast to the examples in Section 4.1, ten new samples that follow the normal distribution $\mathcal{N}(x_i, \sigma_x)$ with $\sigma_x = 0.1$ have been added around each RBF centre.

Figure 4.6 shows the interpolation behaviour for *Thin-Plate*, Gaussian and CSRBF. The Gaussian suffers most from the redundant data problem, even for very small scaling s close to RBF centres, thus strong oscillations ensue. CSRBF shows less sensitive behaviour and a stable interpolation is achieved at large scaling $s = 1$. Furthermore, the predefined continuity C^m of φ plays an important role. The lower the degree m, the more stable is the interpolation even with redundant measurements. From the discussion in the previous section it can be concluded that RBFs with higher smoothness tend to over-fit the data. So to validate the intuitive judgement without performing a large number of experiments, an analysis is presented which provides more general assessment aspects.

From the numerical perspective, a common measure of stability in regression is the *condition number* of the inverted matrix A from (4.3):

$$\text{cond } A = |\frac{\lambda_{max}}{\lambda_{min}}| \tag{4.12}$$

with λ_{max} and λ_{min} as the maximal and minimal eigenvalues of A respectively. The condition increases (becomes worse) when $\varphi(x_k, x_j)$ and $\varphi(x_{k+1}, x_j)$ for neighbouring samples x_k and x_{k+1} are very similar, so the lower eigenvalue decreases. The computation

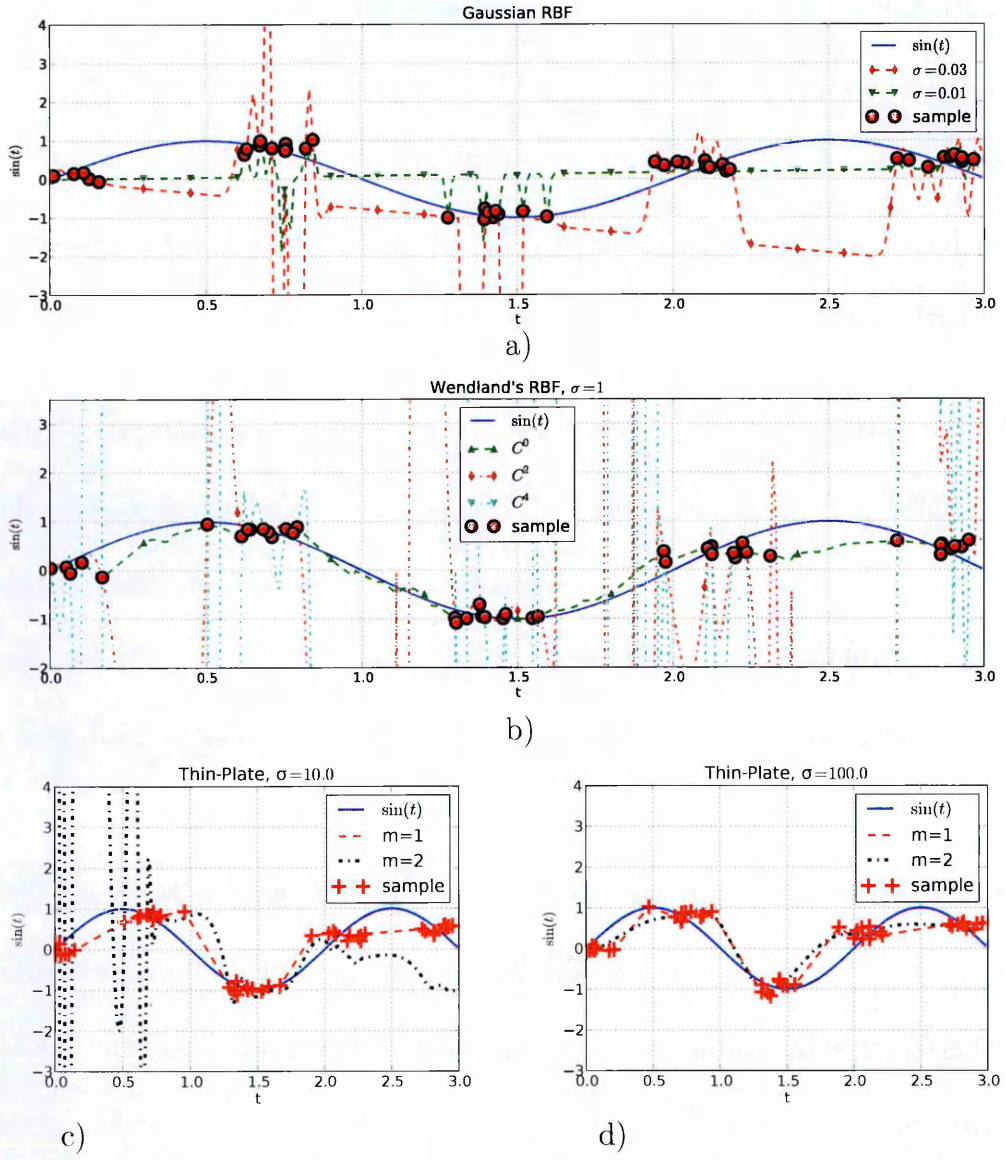


Figure 4.6: Unstable regression with redundant samples. a) Gaussian with scaling $s = 0.01$ and $s = 0.03$, b) Wendland's RBF with $s = 1$ and varying continuity and *c-d) Thin-plate* RBF with varying continuity and scaling.

of the condition A is expensive, so it cannot be applied in applications with large datasets. In addition, no generalization is possible since it depends on the data and small variations in the samples have significant effects. Consequently, this section focuses on a more general stability characterization of the applied RBF.

If two samples have a much closer distance to each other than the scaling parameter

s in the Gaussian or the CSRBF, the condition of A deteriorates. It is thus intuitive to address the minimal distance between two samples or the radius q_x

$$q_x := \frac{1}{2} \min_{j \neq i} \| x_i - x_j \|_2 \tag{4.13}$$

in the dataset. Moreover, the minimal eigenvalue λ_{min} plays a central role in the stability analysis, especially in the field of control theory (Reinschke, 2014). When interpreting $f(x) = \sum_j^M \varphi(x)\alpha_j$ as a transfer function, its behaviour can be analysed in the frequency domain. The key is the Fourier transform of φ. Interpreting the frequency ω as the minimal distance q_x between the interpolated samples allows the best-case stability of the regression model to be estimated analytically. More practically, the lower (worst) boundaries for the lowest eigenvalue are discussed and framed in relation to the expected sample radius q_x. This enables a qualitative assessment of a basis function without performing any numerical experiments. Table 4.2a shows the lower boundaries G for $\lambda_{min} > G(q_x)$, which have been computed according to the Fourier-Bessel transforms and the Borel measure described by Wendland (2004) (p215). The graph in Figure 4.7a clearly shows that the Gaussian is most likely to become unstable (lowest λ_{min}) as also illustrated in Figure 4.6. As further approved by the lower bounds on λ_{min}, the higher the continuity C^m, the higher the probability that a regression system becomes unstable.

Approximation Quality

Another important aspect when selecting a RBF is the best achievable approximation quality. As in stability assessment, numerical experiments often only indicate RBF

RBF	$\lambda_{min} > G(q_x)$
Gaussian	$s^3 e^{-10.63(\frac{3s}{q_x})^2} q_x^3$, $s = 1$
Thin-Plate	q_x^{2m}
CSRBF	q_x^{2m+1}

a)

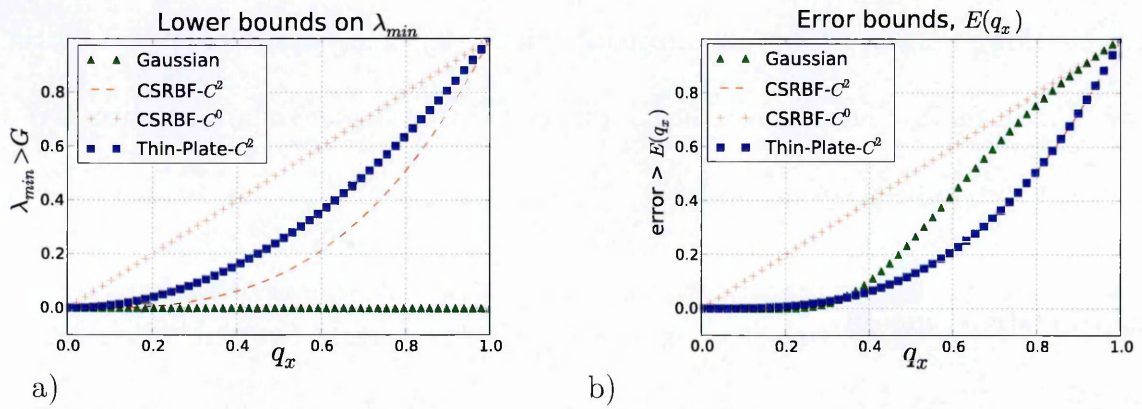
RBF	error> $E(q_x)$
Gaussian	$e^{-s\lvert\log(q_x)\rvert/q_x}$, $s = 1$
Thin-Plate	q_x^{2m}
CSRBF	q_x^{2m+1}

b)

Table 4.2: a) The lower stability bounds (higher is more desirable) for λ_{min}. b) The lower bounds (lower is desirable) for the approximation error of each RBF.

behaviour with respect to one dataset. Fortunately, the theoretical error bounds can be evaluated in a similar way as has been demonstrated with the lower bounds on the minimal eigenvalue of the matrix A. These have been discussed by Wendland (2004) (p188), computed as part of this research and are summarized to provide a complete characterization of the considered RBF. The graph in Figure 4.7b presents the best achievable error for each RBF given the sampling density q_x. The error bounds are shown up to a positive scale factor in Table 4.2b. The lower the error bounds, the higher the approximation quality may be expected from the applied RBF. *Thin-Plate* and CSRBF-C^2 are shown to be superior compared to Gaussian and CSRBF-C^0.

From the evaluation results it is concluded that the higher the sampling density



a) b)

Figure 4.7: a) The lower bounds (higher is more stable) for λ_{min} of different RBFs. b) The lower bounds (lower is better) for the approximation error of each RBF.

(small q_x), the better the approximation quality. Interestingly, the CSRBF-C^2 achieves better quality than other compact RBFs at a lower sampling density q_x and comparable to the *Thin-Plate* RBF. It is important to stress that a small q_x value conflicts with the stability requirements discussed above. A small q_x increases the redundancy in the data and therefore the ability to achieve higher approximation accuracy. However, redundancy also leads to instabilities, which may break the approximation process.

Guiding Principles

From this discussion and taking into account both the stability and shape approximation quality perspectives, the following general guiding principles have been framed for choosing the most appropriate RBF type:

- CSRBF with C^0 is the best choice if the interpolation does not have to be smooth. This is illustrated in Figure 4.7, in which it is evident that in order to achieve the greatest stability, the poorest approximation quality is obtained. Thus, the sampling density needs to be relatively large $q_x \approx 0.1 \cdot s$ with s as the RBF support width.

- If higher degrees are required, e.g. for TV minimization, then *Thin Plate*-C^2 is preferable.

- When the number of RBF centres exceeds $M = 10^4$, CSRBF-C^2 must be applied in order to obtain a result within reasonable time. This particular value of M has been determined empirically.

In choosing the scaling parameter s, it should be set to $q_x =< 0.25s$ which means a single RBF support should include more than four samples in its support. As indicated by the diagrams in Table 4.2 and Figure 4.7, large scaling values ($s \gg q_x$) lead to instability while small ones ($s \ll q_x$) increase the approximation error. So there is a trade-off when selecting the most appropriate value.

These findings have been derived from the general stability and approximation quality estimations shown in Table 4.2 and have been illustrated so far on simple sinusoid signals. Crucially, this result does not depend on the dimension of the data, so the findings are equally valid and applicable to surface approximation of scattered 3D data samples, which will be the main research question addressed in this thesis.

Table 4.3 summarizes the characteristics of the respective RBF types. The more plus signs, the higher the relevance of the RBF regarding the corresponding aspect. The *efficiency* row reflects the sparsity of the constructed kernel matrix which directly influences the computational overhead.

The bottom row of Table 4.3 provides an intuitive summary of the overall performance for each RBF feature, from which a conclusion can be distilled, namely that CSRBF is the preferred choice for shape approximation from scattered 3D samples. While its stability and overall approximation quality are marginally lower than the *Thin-Plate* RBF, its computational efficiency makes it especially attractive for massive 3D datasets. Moreover, the aforementioned stability and the approximation quality issues will be addressed by the integration of regularization strategies, which is the focus of the next section.

4.2 Regularization

This section introduces regularization techniques to deal with regression instabilities. Additionally, regularization enables the integration of specific domain *a priori* knowledge such as smoothness or piecewise planarity of the surfaces. The first discussed regularization technique is the *Tikhonov* regularization. It is one of the oldest but yet most effective approaches initially presented by Tikhonov (1943). The motivation and intuitive reasoning will be presented in Section 4.2.1. Thereafter, sparsity-inducing L_1 based regularization will be introduced. Although statistical analysis via L_1 norms has been discussed for some years in the statistical computing community (Gonin, 1989), the approach only became very popular after a numerical efficient method has been introduced some years ago (Tibshirani, 1994). In contrast to the Tikhonov models, which apply quadratic L_2 terms and lead to global smoothing, L_1 is suitable for piecewise regularization penalties and is therefore of particular interest. The reason is the sparsity-inducing property of the L_1 norm. The shapes are not over-smoothed and sharp edges are preserved when only a few edges and corners are present. This effect on 3D data will be discussed in Chapter 5. In this section, more general regularization with the L_1 norm and its implications on the regression will be discussed critically.

4.2.1 Tikhonov Regularization

Section 4.1 showed that redundant measurements lead to over-fitting and instabilities of a regression process. Motivated by the severe instability common for all regression problems in general, Tikhonov (1943) proposed a technique to improve the condition of equation (4.2) by penalizing the target vector $\boldsymbol{\alpha}$ by $\| \boldsymbol{\alpha} \|_2^2$ additionally to the

cost term. This idea becomes evident when the multiplication of the inverted matrix $A^{-1} = (K^T K)^{-1}$ is applied on $K^T y$. If A has a bad (very high) condition or is even singular, very large values in $\boldsymbol{\alpha}$ will appear due to the division by the smallest eigenvalue. This relation is illustrated in Figure 4.8, in which $\| \boldsymbol{\alpha} \|_2^2$, the Euclidian norm of $\boldsymbol{\alpha}$, is observed when the RBF scale σ is increased. Therefore, minimizing $\| \boldsymbol{\alpha} \|_2^2$ improves the condition of (4.2). The new optimization task is then stated as:

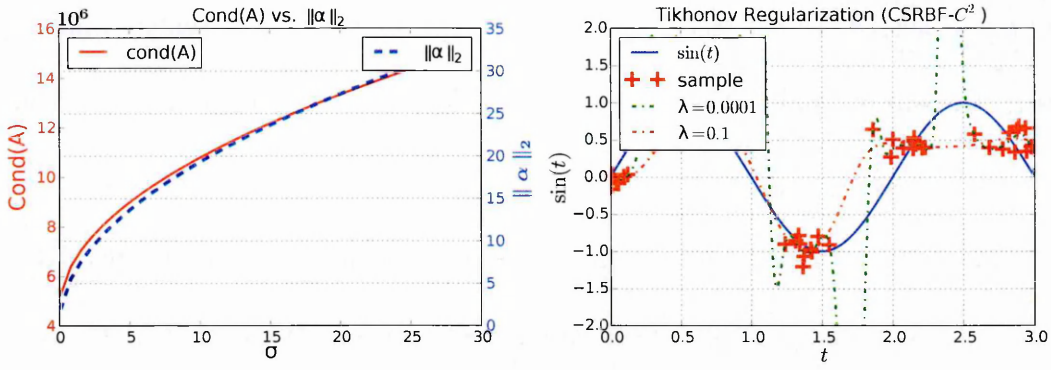
$$\begin{aligned} &\min_{\alpha} \| K\boldsymbol{\alpha} - \mathbf{y} \|_2^2 + \| I\boldsymbol{\alpha} \|_2^2 \\ &\boldsymbol{\alpha}^{opt} = (K^T K + I^T I)^{-1} K^T y. \end{aligned} \tag{4.14}$$

Replacing I with a more general linear operator matrix D enables linear operators to be applied on the solution $\boldsymbol{\alpha}$. Common examples for D are differential operators such as low-pass filters. In discrete models difference or Fourier operators may be represented by D giving preference to specific frequencies (Yang et al., 2010). The drawback of the Tikhonov regularization is that the Euclidian norm $\| D\boldsymbol{\alpha} \|_2^2$ is relatively sensitive to variations in entries of $(D\boldsymbol{\alpha})_i$. For this reason, significant research into efficient optimization models has taken place leading to a common consensus that $\| D\boldsymbol{\alpha} \|_1$, the L_1 penalty term, is more resilient (Hastie et al., 2008; Gonin, 1989). Investigations into these L_1 based regularization techniques will now be reviewed.

4.2.2 Regularization via the Sparsity-Inducing L_1 Norm

From the more generalized perspective, the Tikhonov regularization applies a vector norm

$$l_p = (\sum_{i}^{d} |\alpha_i|^p)^{\frac{1}{p}}$$



a) b)

Figure 4.8: a) Relation between the vector norm $\| \alpha \|_2$ and cond (A). b) Regularization effect with $D = \lambda I$ for two different λ.

with $p = 2$. Such vector norm functions are also known as *Lebesgue-Measures* and have been studied in formal function analysis for many years (Riesz, 1910). It has been proven (Gonin, 1989) that norms with $p < 2$ lead to higher accuracy when applied in model approximation. However, the computational effort usually makes the application intractable in robotics. The L_1 norm received a lot of interest after Chen et al. (1998) presented its convex properties and had demonstrated novel efficient computations, with the idea widely adopted in the signal and image processing domains. Equipped with this new model, improvements in various tasks such as noise suppression, restoration, signal separation, interpolation and extrapolation, compression, sampling, detection, and recognition have been reported (Elad, 2010). The technique is widely known as *Least Absolute Shrinkage and Selection* (Lasso) (Tibshirani, 1994).

The basic effect of L_1 may be enlightened with the so-called *unit circles*, which are also known as the *norm balls* shown in Figure 4.9. Basically, the norm circle is a contour of $\| \boldsymbol{\alpha} \|_p$ along the curve with $\| \boldsymbol{\alpha} \|_p = 1$. Figure 4.9 displays the unit circles of the L_1, L_2 and $L_{0.5}$ norms. The purpose of the diagram is to show the behaviour of the norms, which also can easily be interpreted as cost functions with respect to the two entries in

$\boldsymbol{\alpha}$, α_1 and α_2. If the contour is close to the origin, then it means that the cost putting all values α_i to zero is smaller than when setting only one of them to a value larger than zero. In case of the L_2 norm, the distance to the origin remains constant on a circle. Integrating L_2 into an optimization process means that from the perspective of the overall cost function α_1 or α_2 or both might be preferred. However, when a norm with $p < 2$ is applied, the values of α_1 and α_2 are both reduced to zero (if $p = 0$) when both are expected to be non-zero. Thus, if the application requires making a decision between α_1 and α_2, norms with $p \leq 1$ tend to move the solution to one of the axes α_i since setting all α_i to zero does not lead to a valid solution.

This observation is the key to the sparse data analysis, where the set of $\alpha_i \neq 0$ is expected to be small. Practically, if two RBF centres k^{th} and j^{th} are very close to each other, both weights α_k and α_j will be set to a value close to zero or only one of them will be set to a number larger or smaller than zero. In the case of the L_2 norm, both would receive values larger than in the L_1 case. On the other hand, if a RBF centre is placed on an outlier, then its effect on other weights will be smaller when L_1 is used.

As can be interpreted from Figure 4.9, the $p = 0.5$ norm provides results of higher

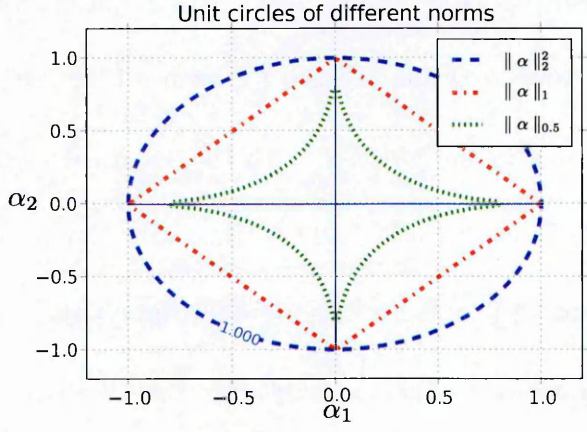


Figure 4.9: The unit circles of the L_2, L_1 and the $L_{0.5}$ norms.

sparsity. However, the optimization becomes non-convex. So reaching a global minimum is not guaranteed.

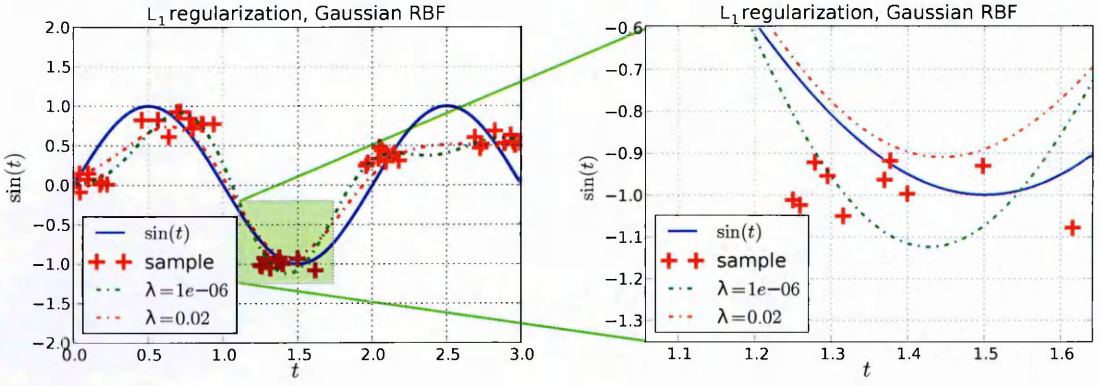
Next, the application of the L_1 norm is compared with the Tikhonov regularization. For the sake of consistency, the interpolation of the signal $\sin(t)$ is reviewed. The new regression task is restated as:

$$\min_{\boldsymbol{\alpha}} \| K\boldsymbol{\alpha} - \mathbf{y} \|_2 + \lambda \| D\boldsymbol{\alpha} \|_1 . \tag{4.15}$$

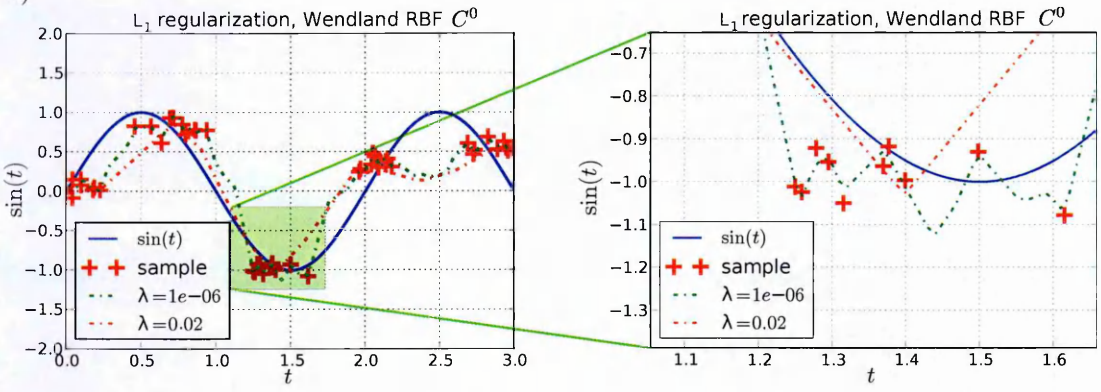
Similar to Section 4.1, the goal is to reconstruct the sinewave signal by adding random samples with a much smaller spacing than the RBF support σ.

Figure 4.10 illustrates the interpolation behaviour of different RBF models as well as values for two different weightings λ. The right column in Figure 4.10a-c shows a zoomed part of the approximation at $t = 1.5$. The dotted curves represent the approximated functions with different weighting parameters λ that controls the strength of the L_1 regularization. The red crosses illustrate the samples.

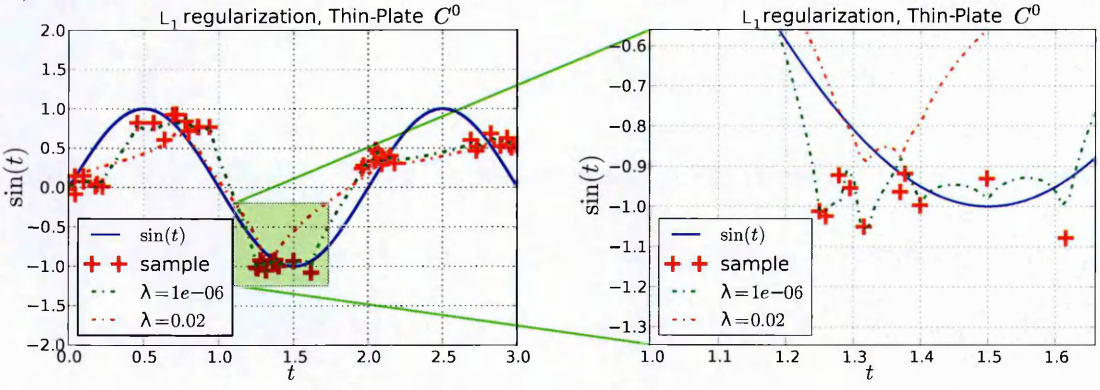
Moreover, Figure 4.10 shows that the Gaussian curve always remains smooth. A strong λ makes the approximation pass *below* the samples (closer to zero). The approximation using CSRBF- and *Thin-Plate-*C^0 RBF reveals more clearly which samples have been dropped. This means that the i^{th} measurement does not contribute to the approximation and the α_i value of a sample at t_i has been set close to $\alpha_i = 0$. An interesting aspect is the stability of the reconstruction. In contrast to the experiments in the previous section, where the smooth RBF models with C^0 continuity have been identified as unreliable in the case of redundant data, the regularization not only stabilizes the



a)



b)



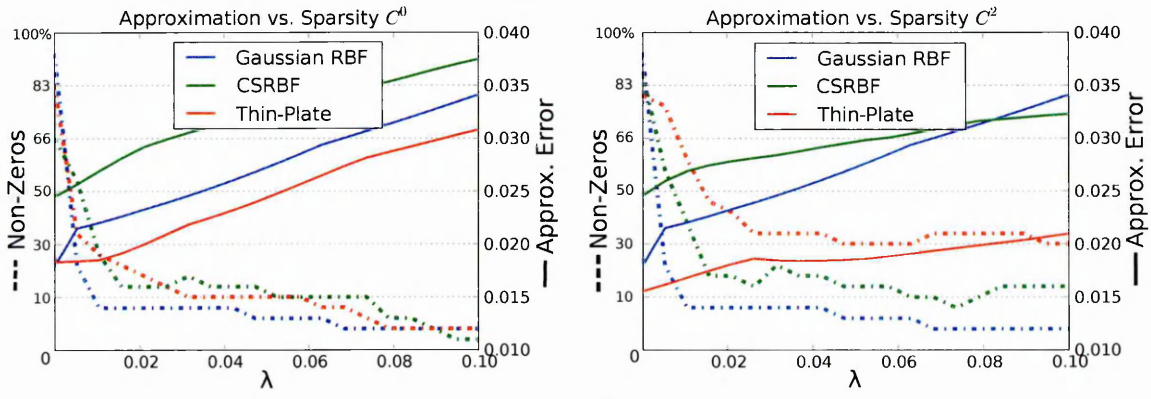
c)

Figure 4.10: L_1 Regularization effect. a) Gaussian RBF, b) CSRBF C^0, c) Thin-Plate RBF C^0 continuity.

regression but also improves the approximation quality significantly.

In this experiment the sparsity weighting λ has been set to two different values to verify and illustrate the effect of the enforced sparsity. This effect is evaluated in more detail in Figure 4.11, where the number of non-zero elements and the approximation quality are elaborated with respect to λ. From the evaluation results in Figure 4.11 it can be concluded that a higher RBF degree allows to increase the sparsity in $\boldsymbol{\alpha}$ without sacrificing the approximation quality. However, the question raised is how to determine the best trade-off controlled by λ. This is addressed in Appendix A in which a standard Cross Validation (CV) technique from Wahba (1990) is described. CV defines a metric which enables the selection of the best from multiple approximations for λ. More recent discussions about the parameter estimation via CV are provided by Zhang et al. (2011).

So far, this section has explained how the L_2 or L_1 regularization affect the approximation on datasets leading to instabilities. Applying the L_1 norm on the parameter $\boldsymbol{\alpha}$, sets the weights α_i of redundant RBF centres to zero. The next Section 4.3 will examine a reliable state-of-the-art technique to solve the L_1 approximation task numerically.



a) b)

Figure 4.11: Effect of λ on the sparsity, a) continuity C^0, b) continuity C^2.

4.3 Efficient Algorithm for Regularized Regression

The algorithm is known as the *alternating direction method of multipliers* (ADMM) (Boyd et al., 2011) and is appropriate to distributed convex optimization, particularly large-scale problems relevant to the main research question. This section introduces the basic form of ADMM before being extended in Chapter 7 into a matrix-free form.

The optimization task is formally stated as

$$\min_{\alpha} f(\boldsymbol{\alpha}) = \frac{1}{2} \parallel \mathbf{y} - K\boldsymbol{\alpha} \parallel_2^2 + \lambda \parallel D\boldsymbol{\alpha} \parallel_1 . \tag{4.16}$$

Initially, the cost function (4.16) is split into two parts: the smooth and convex $f_1(\boldsymbol{\alpha}) = \frac{1}{2} \parallel K\boldsymbol{\alpha} - y \parallel_2^2$ and the non-smooth regularization term $f_2(\boldsymbol{\alpha}) = \lambda \parallel \boldsymbol{\alpha} \parallel_1$. They are further separated into different variables $\boldsymbol{\alpha}$ and $\mathbf{z}$ and coupled together via a constraint:

$$\min_{\boldsymbol{\alpha},\mathbf{b}} f_1(\boldsymbol{\alpha}) + f_2(\mathbf{z})$$

$$\text{s.t. } \mathbf{z} - D\boldsymbol{\alpha} = 0.$$

This constrained optimization problem is restated via the dual decomposition

$$L(\boldsymbol{\alpha}, \mathbf{z}, \mathbf{b}) = f_1(\boldsymbol{\alpha}) + f_2(\mathbf{z}) + \mathbf{b}^T(D\boldsymbol{\alpha} - \mathbf{z})$$

which is known as the Bregman-Split form from Goldstein and Osher (2009). In order to improve the convergence of the method, the *augmented Lagrangian* published by

Hestenes (1969) is superimposed to give

$$L_\rho(\boldsymbol{\alpha}, \mathbf{z}, \mathbf{b}) = f_1(\boldsymbol{\alpha}) + f_2(\mathbf{z}) + \mathbf{b}^T(D\boldsymbol{\alpha} - \mathbf{z}) + \rho/2 \parallel D\boldsymbol{\alpha} - \mathbf{z} \parallel_2^2 \tag{4.17}$$

with $\rho > 0$ as the constraint controlling weight. So on every iteration, the three following steps have to be performed:

1. $\boldsymbol{\alpha}^{k+1} := \min_{\boldsymbol{\alpha}} L_\rho(\boldsymbol{\alpha}, \mathbf{z}^k, \mathbf{b}^k)$

2. $\mathbf{z}^{k+1} := \min_{\mathbf{z}} L_\rho(\boldsymbol{\alpha}^{k+1}, \mathbf{z}, \mathbf{b}^k)$

3. $\mathbf{b}^{k+1} := \mathbf{b}^k + (D\boldsymbol{\alpha}^{k+1} - \mathbf{z}^{k+1})\rho$

In explicit form, the three steps defined above are:

ADMM for L_1 approximation:

1. Solve for $\boldsymbol{\alpha}$: $(K_n^T K_n + K^T K + \rho D^T D)\boldsymbol{\alpha} = K_n^T \mathbf{n} + D^T(\rho \mathbf{z} - \mathbf{b})$

2. Solve for each z_i separately:

$$\min_{z_i} \lambda |z_i| + \mathbf{b}^T(D\boldsymbol{\alpha} - \mathbf{z}) + \frac{\rho}{2}(\boldsymbol{\alpha}^T D^T D\boldsymbol{\alpha} - 2\mathbf{z}^T D\boldsymbol{\alpha} + \mathbf{z}^T \mathbf{z})$$

3. Evaluate: $\mathbf{b}^{k+1} := \mathbf{b}^k + (D\boldsymbol{\alpha}^{k+1} - \mathbf{z}^{k+1})\rho$

So far, the general form of ADMM has been extended to the case of L_1 regularization applying the matrix D. Note that D may be a differential operator enabling piecewise penalization of areas with high curvature. If D is the identity matrix, then it is a

simple smoothing operator, known as Lasso. While steps 2 and 3 can be executed in parallel with complexity of $\mathcal{O}(N)$, the computation time of step 1 usually suffers from complexity $\mathcal{O}(N^3)$. For datasets with more than $N > 10^4$ points, the computation becomes intractable for robotics applications (Bach et al., 2012). This issue will be further addressed in Chapter 6, where a fast iterative solution based upon the Gauss-Seidel scheme will be developed.

4.4 Discussion

When performing signal approximation with RBF, several crucial aspects have been identified:

- The approximation quality increases with the smoothness of RBF. E.g. C^2 leads to a higher approximation quality than C^0-RBF (see Figure 4.2).
- When the smoothness is increased, the stability of the optimization task is reduced (see 4.6b).
- Higher RBF scales reduce the stability but lead to more successful noise suppression. Thus the trade-off between good noise suppression and stability needs to be maintained (see 4.6c-d).
- Regularization helps to deal with redundant samples and thus increases the stability, so larger RBF scaling values may be applied without reducing the stability (see Figure 4.8b).
- The sparsity domain can be controlled by the matrix D, which is an identity Matrix when Thikhonov regularization is applied (see (4.14)). When performing

edge aware shape reconstruction, $D\boldsymbol{\alpha}$ will represent the variation of the shape. The regularization term $\| D\boldsymbol{\alpha} \|_1$ will increase the cost function on shapes with high variation such as edges or corners.

- When extending the approximation task with a regularization term, the best sparsity effect is achieved when applying *Thin-Plate* or CSRBF with C^2 (or higher) smoothness (see Figure 4.11).

- The algorithm (ADMM) presented to solve the L_1 regularization task numerically is flexible enough to deal with any positive-semidefinite matrix D. Its first step, however, relies on dense algebra computation and usually suffers from high computational overhead. This issue will be solved in Section 6.

- CSRBF has been selected for the integration in the 3D shape approximation task because of its favourable smoothness, stability and the effect on the computation overhead (see Table 4.3). The plus signs for each criteria have been derived from the previous discussion in Section 4.1.2. The efficiency of a RBF relates to the ability to solve the first step of ADMM when the matrix K is constructed using the particular RBF. The insights provided in this chapter will help to define the 3D shape approximation as a combination of CSRBF, L_1 regularization over $D\boldsymbol{\alpha}$ and the application of ADMM.

Table 4.3: Comparative overview of the RBF models

	Gaussian	*Thin Plate*	CSRBF
Stability		+ + +	+ + +
Approximation	+ +	+ + +	+ + +
Smoothness	+ + +	+ + +	++
Efficiency	+		+ +
Σ	6	9	10

4.5 Summary

This chapter set up the theoretical foundation for efficient interpolation with radial basis functions. The elaboration results of suitable RBFs for general approximation, the RBF selection criteria based on stability and approximation quality have been provided. The stability has further been addressed in the discussion on regularization techniques framed by a convex optimization task. A numerical method for solving the RBF regression problems with L_1 regularization operators (matrix D) has been developed and presented. The illustrative discussions provided in this chapter aim to support the underlying theory applied in Chapters 5 and 6, where 3D shape reconstruction on synthetic and realistic datasets will be performed. The next chapter will present the application of these fundamentals for 3D shape approximation from scattered 3D samples.

5 Shape Reconstruction from Scattered 3D Points

This chapter presents novel solutions to address both the first (shape representation) and second (planarity regularization) research objectives defined for the knowledge integration framework presented in Figure 1.3. The shape of the sampled surface is reconstructed by using a set of scattered 3D points as input. In order to achieve this, the shape is modelled as an implicit function which is defined as a weighted sum of RBFs. The first research objective is fulfilled by extracting the weights from a convex cost functional leading to a consistent and smooth shape. To fulfil the second research objective of the new framework, the *a priori* piecewise planarity assumption is incorporated by extending the optimization process using TVL_1 regularization (Section 5.2). Quantitative evaluation upon a synthetic dataset has shown that TVL_1-based shape approximation achieves higher accuracy compared with existing approaches including LSQ, Lasso and TVL_2.

In particular, in Section 5.1 the underlying shape regression model is established. Section 5.2 will discuss the effect of regularization techniques such as Lasso and TVL_1 on different RBF types and elaborates the approximation quality depending on involved parameters such as the RBF scale or the regularization term weighting. In this chapter simple synthetic data is used to evaluate and to validate the algorithm. The application of the algorithm on realistic datasets will be discussed in Chapter 7.

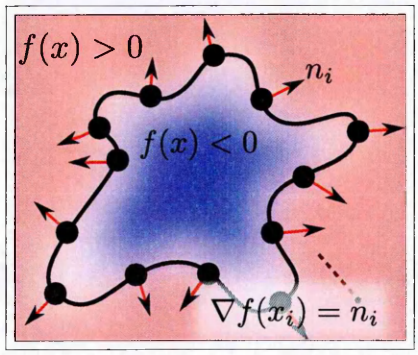


Figure 5.1: Implicit shape function $f(\mathbf{x})$, samples (dots) and orientation vectors $\mathbf{n}_i$. The colours illustrate the value of $f(x)$. Red: $f(x) > 0$, blue: $f(x) < 0$.

5.1 Implicit Shape Regression

The principal idea of shape modelling with RBF is to extract an implicit function that represents the shape of interest by its zero values. Figure 5.1 shows an implicit function $f(\mathbf{x})$ with colour-encoded values. All segments where the function is zero are considered as part of the shape of interest and are highlighted with a thick line. The task is the approximation of f from the given set of sample points (thick dots) and its available orientation vectors $\mathbf{n}_i$.

More formally, an algebraic function $f(\mathbf{x}), f : \mathbb{R}^3 \mapsto \mathbb{R}$ which is defined in the 3D space needs to be reconstructed. In Figure 5.1 the curve is representing the space of $\mathbf{x}$ with $f(\mathbf{x}) = 0$. Given a set of measured 3D points $\mathbf{x}_i$, this condition defines a constraint on each sample point $f(\mathbf{x}_i) = 0$. Thus, f must be zero at each i^{th} sample $\mathbf{x}_i$ and has to interpolate between these samples. The surface normals at every sample are employed since the zero level alone does not provide information about the orientation of the surface. The task is stated as a search for f giving zero at each sample position $f(\mathbf{x}_i) = 0$ *and* $\nabla f(\mathbf{x}_i) = \mathbf{n}_i$ approximating the normals by the gradient of f. By integrating all this information, the convex regression task is formulated with the objective of

minimizing the cost function:

$$\min_f \sum_i^N \| f(\mathbf{x}_i) \|_2^2 + \| \mathbf{n}_i - \nabla f(\mathbf{x}_i) \|_2^2 \tag{5.1}$$

where $\nabla f(\mathbf{x}_i)$ is the gradient with respect to $\mathbf{x}$ of f at $\mathbf{x}_i$. The length constraint of the normals $\| \nabla f(\mathbf{x}_i) \|_2 = 1$ is omitted in order to simplify the optimization problem. Following the basic RBF interpolation concepts presented in Chapter 4, the non-linear function f is defined as a linear combination of M distinct RBF as

$$f(\mathbf{x}) = \sum_m^M \varphi_m(\mathbf{x}_m, \mathbf{x})\alpha_m.$$

Integrating this into the regression of N samples, the cost function becomes

$$\min_{\boldsymbol{\alpha}} \sum_i^N \left[(\sum_m^M \varphi_m(\mathbf{x}_m, \mathbf{x}_i)\alpha_m)^2 + \| \mathbf{n}_i - \sum_m^M \nabla_x \varphi_m(\mathbf{x}_m, \mathbf{x}_i)\alpha_m \|_2^2 \right]. \tag{5.2}$$

In order to obtain the gradient ∇f, only the gradient of φ needs to be computed, which is available in analytical form (see Appendix C). Rewriting (5.2) in matrix notation leads to a shorthand version of the cost function:

$$\min_{\boldsymbol{\alpha}} \| K\boldsymbol{\alpha} \|_2^2 + \| \mathbf{n} - K_\nabla \boldsymbol{\alpha} \|_2^2 . \tag{5.3}$$

The matrix K contains the RBFs $K_{n,m} = \varphi(\mathbf{x}_m, \mathbf{x}_n) \in \mathbb{R}$ and $K_{\nabla n,m} = \nabla\varphi(\mathbf{x}_n, \mathbf{x}_m) \in \mathbb{R}^3$ as introduced in Chapter 4. The matrices are of the dimensions $K \in \mathbb{R}^{N \times M}$ and $K_\nabla \in \mathbb{R}^{3N \times M}$. All α_i are summarized in one vector $\boldsymbol{\alpha}$. Note that RBFs with local

support return zero for the distant point pairs $\mathbf{x}_n, \mathbf{x}_m$ and lead to sparse matrices K and K_∇, which improves the storage requirement and computational efficiency. This is because whenever $\mathbf{x}_n$ and $\mathbf{x}_m$ are far away from each other, $\varphi(\mathbf{x}_n, \mathbf{x}_m)$ and CSRBF are applied.

Since (5.3) has the usual LSQ form, it can be rewritten as

$$\min_{\boldsymbol{\alpha}} \boldsymbol{\alpha}^T (K^T K + K_\nabla^T K_\nabla)\boldsymbol{\alpha} - 2\mathbf{n}^T K_\nabla \boldsymbol{\alpha} \tag{5.4}$$

and solved in one linear algebra step:

$$\boldsymbol{\alpha} = A^{-1} K_\nabla^T \boldsymbol{n}, \text{ with } A = K^T K + K_\nabla^T K_\nabla. \tag{5.5}$$

Note that matrix inversion is usually a computing power demanding operation and is not practical for matrices A larger than e.g. 100×100. However, since $K^T K$ and $K_\nabla^T K_\nabla$ are positive definite, so is A. Thus, the more efficient Cholesky factorization of A can be performed, which enables to reduce the computational effort.

The *Thin-Plate* RBF requires an additional polynomial in order to guarantee a valid solution as discussed in Chapter 4. When applying the *Thin-Plate* RBF, f becomes $f(\mathbf{x}) = \sum_m^M \varphi(\mathbf{x}_m, \mathbf{x}) + p(\mathbf{x})^T \mathbf{c}$ with $p(\mathbf{x})$ as the additional polynomial of degree $m-1$ (Wahba, 1990). So the matrices K and K_∇ from (5.3) and the target parameter vector $\boldsymbol{\alpha}$ are extended to

$$\tilde{K} = \begin{pmatrix} K & B \\ B^T & \mathbf{0} \end{pmatrix}, \; \tilde{K}_\nabla = \begin{pmatrix} K_\nabla & B_\nabla \end{pmatrix}, \; \tilde{\boldsymbol{\alpha}} = \begin{pmatrix} \boldsymbol{\alpha} \\ \mathbf{c} \end{pmatrix} \tag{5.6}$$

with

$$B = \begin{pmatrix} p(\mathbf{x}_1)^T \\ \dots \\ p(\mathbf{x}_N)^T \end{pmatrix} \text{ and } B_\nabla = \begin{pmatrix} \nabla \mathbf{p}(\mathbf{x_1})^\mathbf{T} \\ \dots \\ \nabla \mathbf{p}(\mathbf{x_N})^\mathbf{T} \end{pmatrix} \tag{5.7}$$

where $\nabla \mathbf{p}(\mathbf{x_i})^\mathbf{T}$ is the gradient of the polynomials. Having applied these modifications, the LSQ is solved as stated in (5.5).

In order to visualize the implicit geometric shape functions, the border between positive $f(x) > 0$ and negative $f(x) \leq 0$ implicit values is extracted and rendered as a mesh via OpenGL. This process is illustrated in Figure 5.2 and is known as the *Marching Cubes* algorithm (Lorensen and Cline, 1987).

Figure 5.3 demonstrates the LSQ solution on a synthetic ideal 3D step function that contains neither noise nor outliers. Figures 5.3c and 5.3d show the reconstructed 3D shapes using the *Thin-Plate* and CSRBFs of degree C^2. Figure 5.3b illustrates positive values of f by a green colour and negative values by a red one. Image pixels where f attains zero are highlighted in white. The image has been generated along the slicing-plane as shown in Figure 5.3a highlighted in red. The 3D samples are shown as black dots.

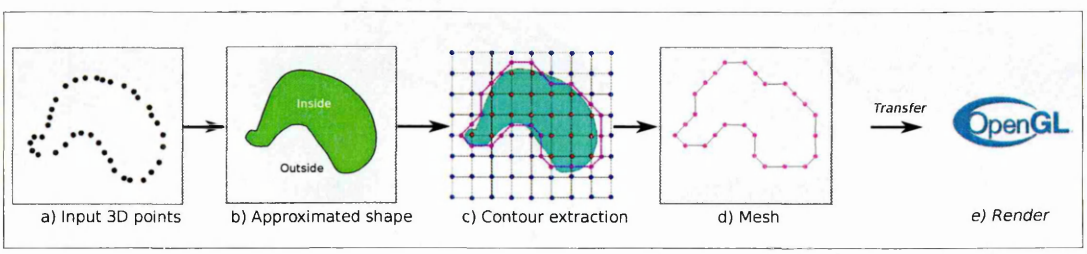


Figure 5.2: Scattered samples (a) are processed to an implicit shape (b), processed to a mesh (c,d) via Marching Cubes and visualized via OpenGL (e).

The results obtained from *Thin-Plate* (Figure 5.3c) and CSRBF (Figure 5.3d) are similar since the input dataset is not perturbed by noise or outliers. This, however, will change when error-prone samples are processed, which will be the focus of Section 5.2. Note that the implicit shape is also defined outside of the sampled area since the general RBF-based approximation concept is well-suited for inter- and extrapolation. Also note that the shapes are smooth outside of the sampled area, which is an indicator for a stable approximation.

The following Section 5.2 will introduce regularization which improves the sensibility to noise and outliers as well as the stability of the shape approximation.

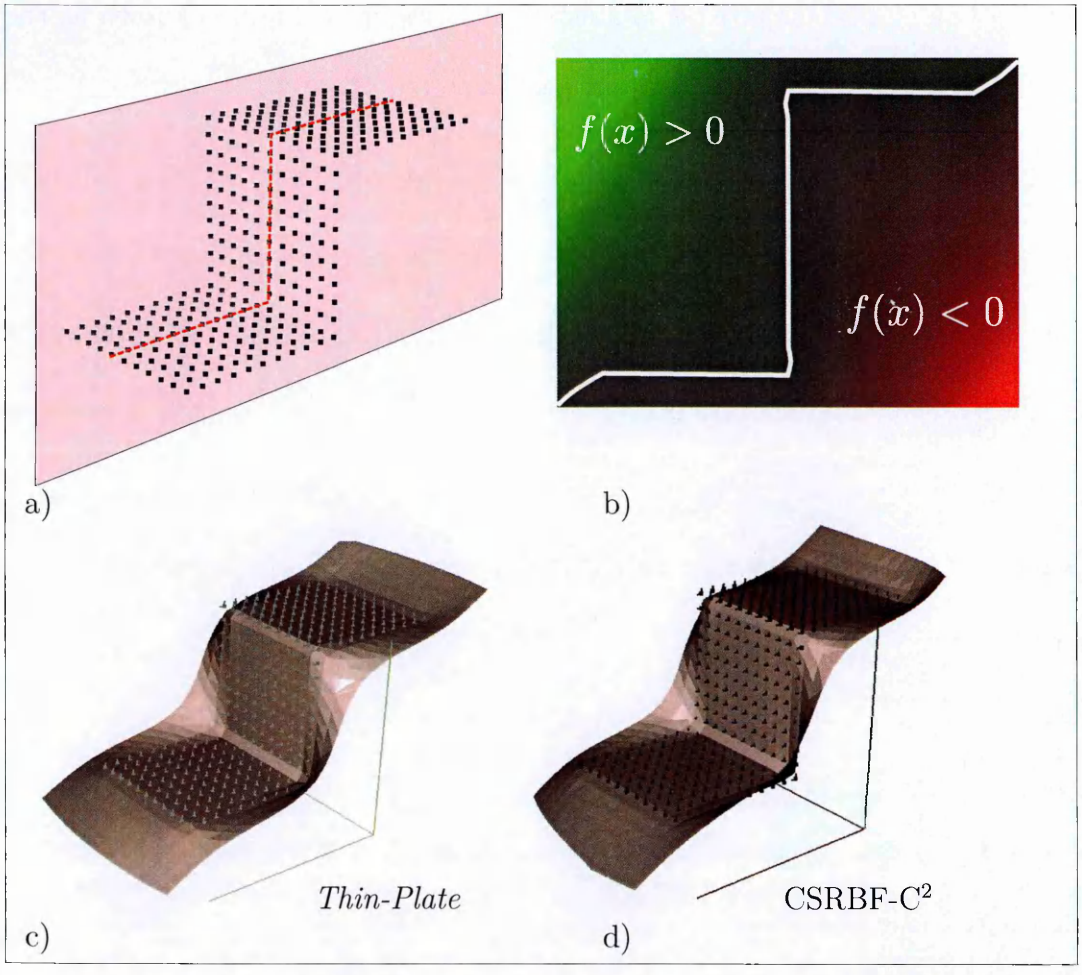
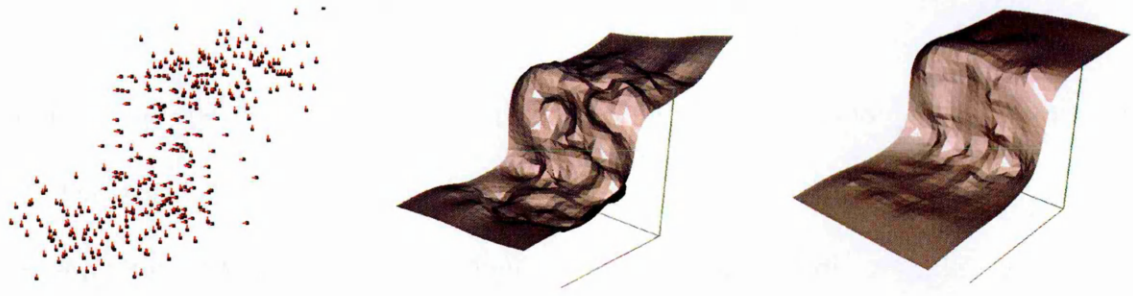


Figure 5.3: a) The input dataset. b) The implicit function along the slice-plane (*Thin-Plate* RBF). c) Rendered *Thin-Plate* RBF and d) CSRBF C^2 reconstruction.



a) b) c)

Figure 5.4: a) Noisy 3D samples of the step function. b) Direct LSQ approximation. c) Regularized (Lasso) approximation.

5.2 Noise Suppression via Regularization

3D samples acquired from a sensor generally contain errors and outliers, though these should not perturb the final geometry reconstruction result. This section introduces three proposed regularization techniques which significantly enhance the robustness of the 3D shape approximation process by both reducing the sensitivity to noise and increasing the stability of the RBF-based LSQ regression.

Figure 5.4 shows an example in which a) is the noisy input dataset, b) the reconstruction via the unregularized LSQ method, and c) a result obtained via Lasso regularization. As expected from the discussion in Chapter 4, the regularization technique improves the accuracy of the reconstruction from noisy data. Now, this will be elaborated in detail with Lasso and two variants of total variation minimization approaches.

5.2.1 Lasso Regularization

The Lasso technique was initially proposed for sparsity-inducing regularization by Tibshirani (1994). The technique applies the L_1 norm on the parameter vector $\boldsymbol{\alpha}$, which further improves the stability and robustness to noise. Having placed the RBF

centres directly on the samples, this may lead to redundant RBFs and cause unstable approximation, as pointed out in Chapter 4. This issue is, however, solved by the L_1 regularization. If the *m-th* basis function $\varphi(\mathbf{x}_m, \mathbf{x})$ does not contribute to the solution, it is deactivated by a zero weight α_m. Furthermore, the L_1 regularization suppresses outliers and error measurements efficiently.

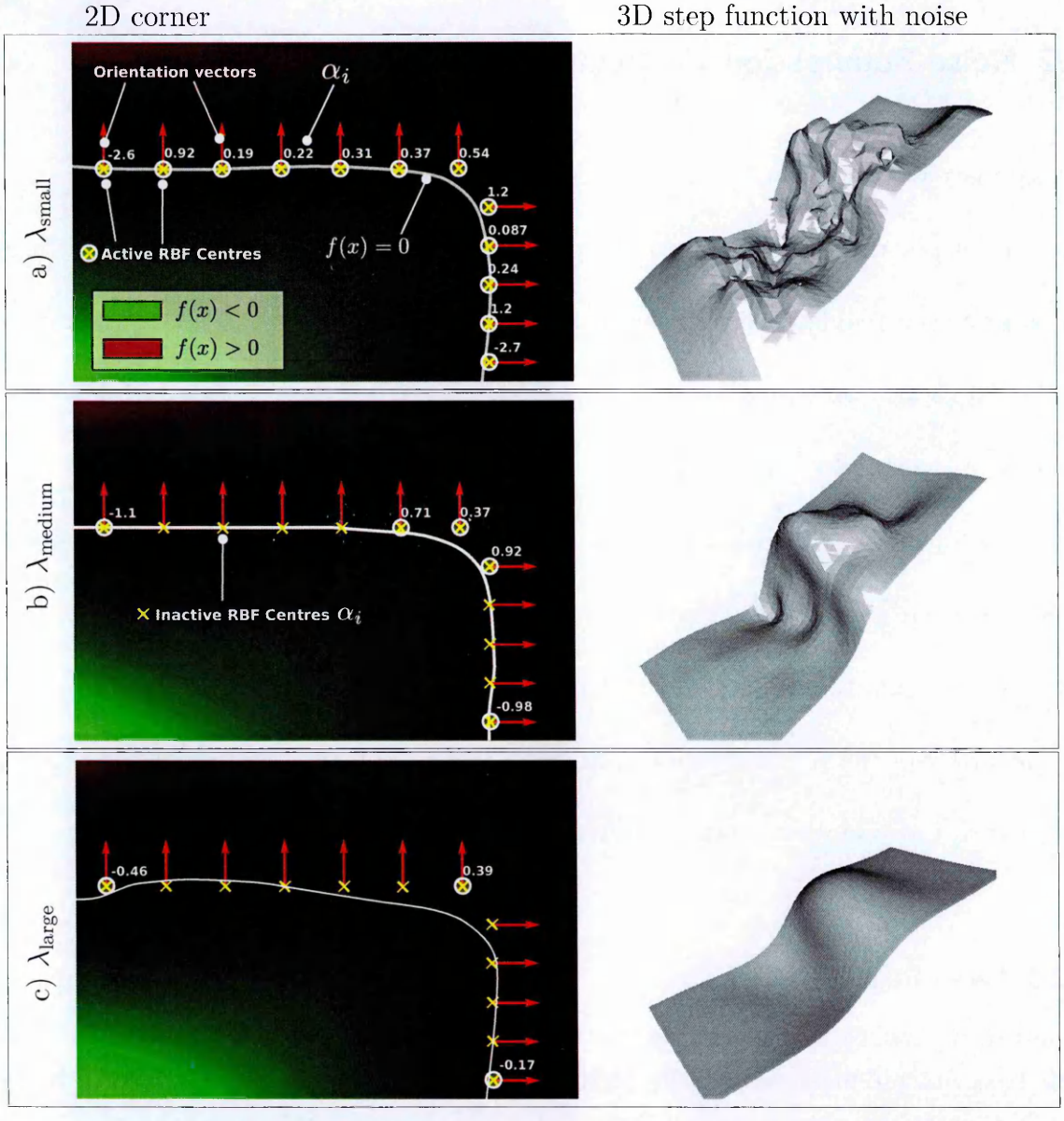


Figure 5.5: Lasso regularization on a noise-free 2D corner and the error-prone 3D step function. a) Shape approximation with small λ, b) medium λ and c) large λ leading to strong over-smoothing.

When extending the Lasso form to the underlying RBF-based regression, the cost function is restated as

$$\min_{\boldsymbol{\alpha}} \| K\boldsymbol{\alpha} \|_2^2 + \| \mathbf{n} - K_{\nabla}\boldsymbol{\alpha} \|_2^2 + \lambda \| \boldsymbol{\alpha} \|_1 \tag{5.8}$$

where the weighting parameter λ controls the impact of the regularization. Figure 5.5 illustrates the effect of the Lasso model on 2D and 3D approximations. In the left column circles illustrate the RBF centres that have non-zero weights $\alpha_i \neq 0$. The numbers close to them show the α_i values obtained after minimizing equation (5.8). For a more insightful illustration explanation, the 2D examples on the left column do not include any noise. The short red lines indicate the orientation of each sample (yellow cross). The resulting shape is shown as a white line. In Figure 5.5b a specific λ is reached where the shape is very similar to the case when all RBF centres are employed, but only a limited set of RBFs remains active receiving a weight $\alpha_i \neq 0$. The generalized cross validation technique from Wahba (1990) (Appendix A) has been applied in order to find this condition. Even though the regularization with Lasso improves the approximation quality, it does not distinguish between error-prone samples on corners and samples along planar areas. Motivated by the fact that areas along planar segments should receive stronger smoothing than along edges, total variation minimization is integrated into the shape regression, which will be discussed next.

5.2.2 Total Variation - L_2

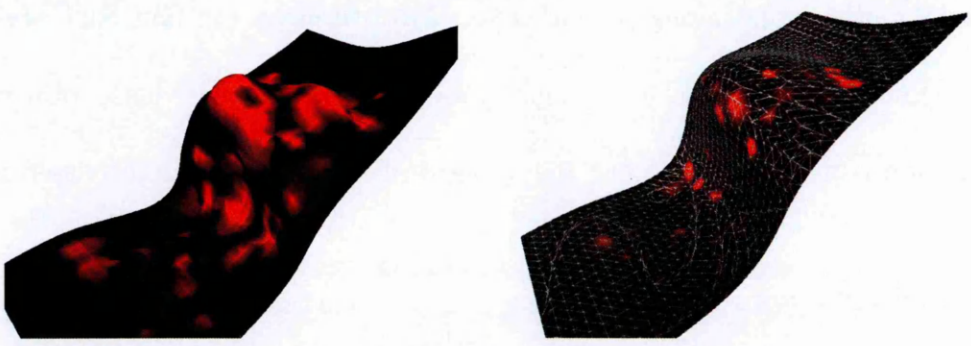
The TV technique is motivated by the fact that most shapes encountered in real-world environments are planar in nature. The novelty in this proposed solution is to exploit

this knowledge by a regularization operator that gives high penalties at corners and edges and lower penalty costs in planar areas.

Planar areas share a common property: the second derivative of the shape function is zero. Thus, the goal is to define a regularization term that penalizes the intensity of the second derivatives along a shape.

Figure 5.6 illustrates the TV cost on an extracted shape (Figure 5.3) reconstructed from error-prone samples in which the intensity of the second derivatives is encoded in red colour. Figure 5.6a illustrates a shape where the TV has been calculated without changing its geometry. Figure 5.6b shows the shape after optimization where the TV has been jointly minimized via regularization. Qualitatively, the introduction of the TV penalty has improved the corners and the planarity of the extracted shape leading to a more accurate reconstruction. A quantitative validation of these findings will be presented in Section 5.2.5.

Optimization models employing the TV regularization are well-known in the discipline of variational optimization in computer vision. It is usually performed by computing the Hessian matrix of the target function. In contrast to this state-of-the-art methodology,



a) b)

Figure 5.6: The effect of TV minimization. a) The TV cost is displayed in red colour on the extracted shape. b) The cost has been employed to improve the shape reconstruction.

the second derivative is obtained by performing derivatives with respect to the radius r of the RBF $\varphi(r)$. Compared to the evaluation of the 3×3 Hessian matrix, this reduces the computational cost by a factor of nine and is easier to compute analytically (see Appendix C). Similar to computing the gradients of f, the second derivative is also a sum of derived RBFs. These are obtained via

$$TV(\mathbf{x}) = \sum_{m}^{M} \partial_{rr}^2 \varphi(r) \alpha_m$$

with $r = \| \mathbf{x}_m - \mathbf{x} \|_2$. Explicit second order derivatives of the RBFs are presented in Appendix ??. Applying the TV regularization, the cost term (5.8) is extended by D to

$$\min_{\boldsymbol{\alpha}} \| K\boldsymbol{\alpha} \|_2^2 + \| \mathbf{n} - K_\nabla \boldsymbol{\alpha} \|_2^2 + \lambda \| D\boldsymbol{\alpha} \|_2^2 \tag{5.9}$$

with $D_{i,m} = \partial_{rr}^2 \varphi(\mathbf{x}_i, \mathbf{x}_m)$. Any positive semi-definite matrix D can be used for regularization purposes as stated by Smola and Schölkopf (1998). The factors α_m corresponding to the largest eigenvalue of D are attenuated most while weights which lie within the kernel of D are not reduced at all. The solution is obtained via a single linear algebra step:

$$\boldsymbol{\alpha} = A^{-1} K_\nabla^T \boldsymbol{n}, \text{ with } A = \lambda D^T D + K^T K + K_\nabla^T K_\nabla.$$

A qualitative and quantitative evaluation of the TVL_2 approach will be discussed in Sections 5.2.4 and 5.2.5 respectively. The following section introduces the TVL_1 approximation model.

5.2.3 Total Variation - L_1

The TVL_1 model is developed in order to preserve sharp edges and to increase the robustness of the shape approximation. In contrast to the L_2 norm $\| D\boldsymbol{\alpha} \|_2$, this extension applies the L_1 norm $\| D\boldsymbol{\alpha} \|_1$. The minimized cost function is now defined as

$$\min_{\boldsymbol{\alpha}} \| K\boldsymbol{\alpha} \|_2^2 + \| \mathbf{n} - K_\nabla \boldsymbol{\alpha} \|_2^2 + \lambda \| D\boldsymbol{\alpha} \|_1 .$$

Now, the qualitative and quantitative evaluations of the presented LSQ, Lasso TVL_2 and TVL_1 approaches will be compared.

5.2.4 Qualitative Approximation Performance

Figure 5.7 shows the shape approximation results for the noisy 3D step function processed by the proposed Lasso, TVL_2 and TVL_1 techniques. In addition, different RBF types have been applied in shape reconstruction. The red colour in the images reflects the TV cost estimated of the shape. The black surfaces for the approximated shapes using the Gaussian RBF are due to the Gaussian function being smoother than CSRBF, which leads to lower variations. However, the CSRBF shape still has sharper edges. These have been smoothed out in the Gaussian case. Since the *Thin-Plate* basis functions are designed to provide smallest deviations of the reconstructed shape, the TV cost is much smaller than for CSRBF or Gaussian. This also leads to smoother results in planar areas. Note that the shape reconstructed by Gaussian tends to curl at the extrapolated edges, which is the result of instability. As proposed in Section 4.1.3, if not stated differently, for all experiments the same RBF scale of $s/q_x = 4$ units

has been applied enclosing approximately 16 samples. The implementation details of the RBF centres' distribution are described in Appendix E. The weighting parameter λ for Lasso has been estimated via cross validation to $\lambda_{lasso} = 10^{-3}$, $\lambda_{TV} = 0.5$. The influence of the RBF scale parameter s will be elaborated next.

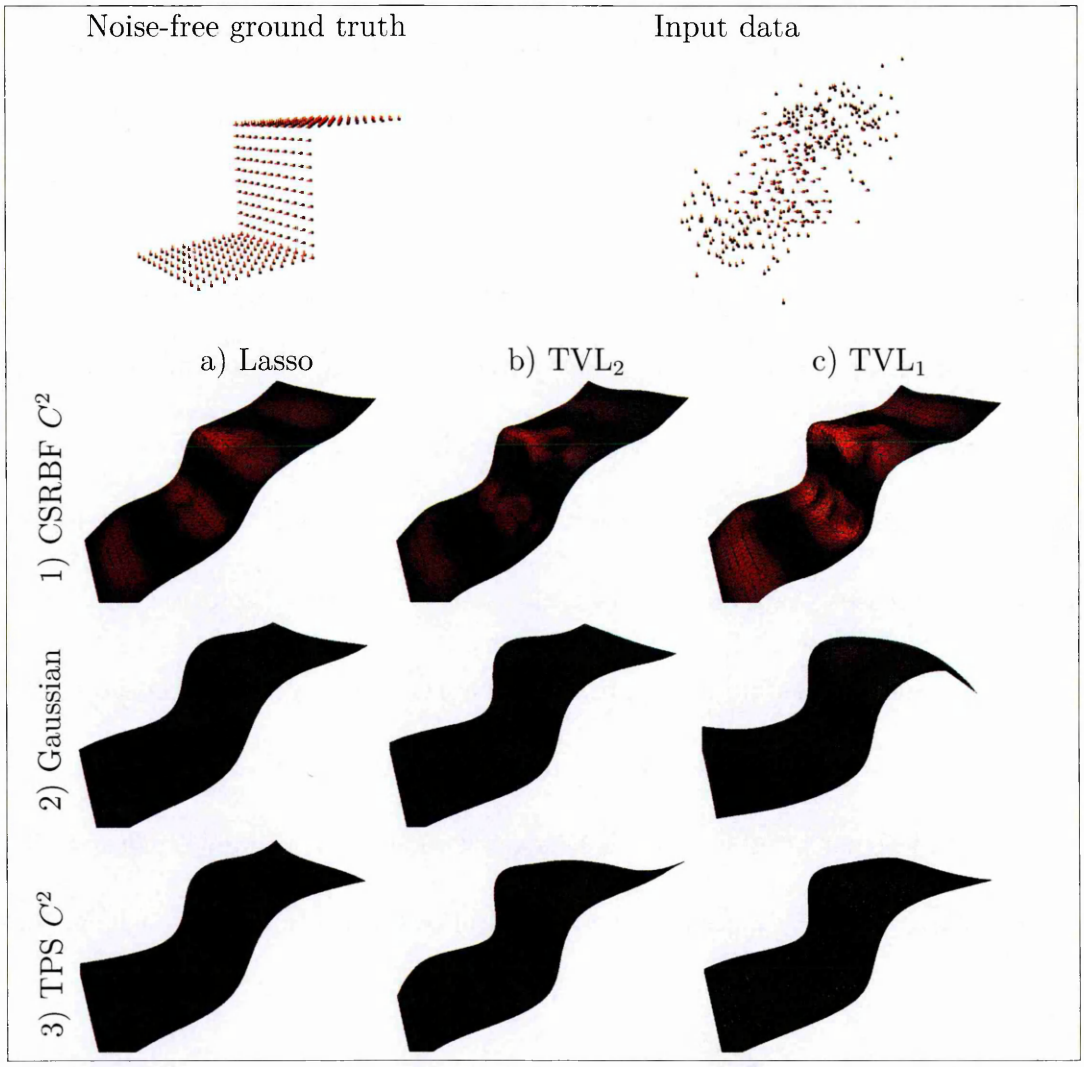


Figure 5.7: Approximation results from error-prone samples with Lasso (a), TVL_2 (b) and TVL_1 (c) using 1) CSRBF-C^2, 2) Gaussian and 3) *Thin-Plate* RBF. The TV intensity cost is shown in red.

5.2.5 Quantitative Approximation Performance

The RBF scaling parameter is crucial for the overall shape approximation quality. If the RBF supports do not overlap, no approximation is possible except in the case of the *Thin-Plate* RBF. Large values s make the resulting approximation too smooth by removing edges and corners. The following analysis seeks to critically assess the impact of scaling. The *Thin-Plate* and CSRBF have been set to C^2 since TV computations require a smoothness degree of at least two. Unregularized LSQ approximation has also been applied for comparison.

As illustrated in Figure 5.7, the positions and orientations of the ground truth samples are perturbed with Gaussian noise of $\sigma \approx 30\%$ of the object size. Note that this noise level notably exceeds the usual level applied in 3D shape reconstruction evaluations. Most approaches that inject noise have 1% to 5% of the object size (Fleishman et al., 2005). This also indicates the increased robustness of the proposed solution.

The precise ground truth points are projected on $f(\mathbf{x})$ after $f(\mathbf{x})$ was successfully reconstructed. The projection is performed along the gradient direction $\nabla f(\mathbf{x}_{gt})$ at each of the ground truth points. The distance between each ground truth sample and the zero level of f is considered as the deviation of the precise shape. Figure 5.8

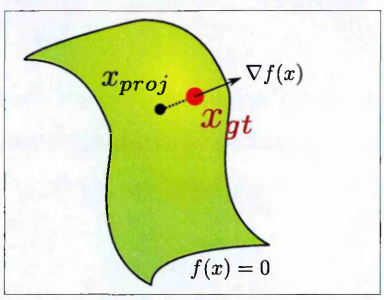


Figure 5.8: The distance between x_{gt} and its projection on f, x_{proj}.

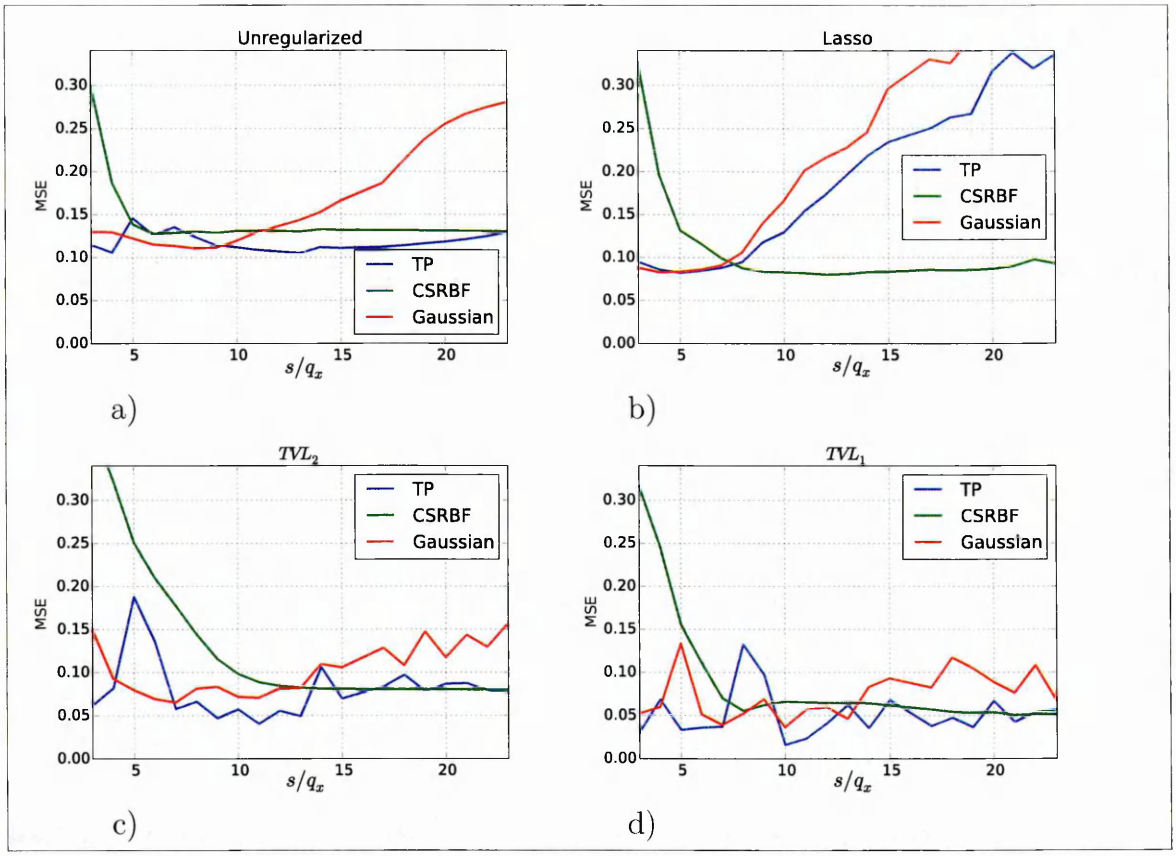


Figure 5.9: Projection MSE for a) unregularized LSQ, b) Lasso, c) TVL_2, d) TVL_1 strategy.

illustrates the projection of x_{gt} onto the surface of f. The deviation is then calculated as $\| x_{proj} - x_{gt} \|_2$ and the overall MSE is determined as the approximation quality metric.

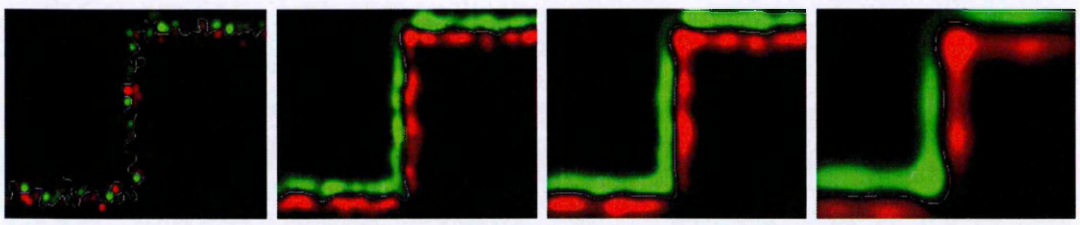
Figure 5.9 shows the achieved MSE for each regression strategy with a varying scaling parameter s in terms of the sampling density q_x. For example, $s/q_x = 4$ means that the RBF support is four times larger than the sampling density of the dataset. Similar to the elaboration given in Chapter 4, the approximation using RBF achieves a best accuracy of MSE=0.12 at $s/q_x = 7$. TVL_1 (Figure 5.9d) achieves the highest accuracy (lowest MSE value) in all four cases. The MSE is approximately 10% lower compared to TVL_2 or the Lasso approach. Moreover, large scaling s lead to over-smoothing and decreasing approximation quality. However, this situation is handled successfully by

both TV regularization techniques.

Figure 5.10 shows the effect of increased support s when the TVL_1 method is applied. If s/q_x is too small, the reconstructed shape has a strong curvature. Larger scaling values, however, lead to an increasing smoothness of the shape and higher attenuation of strong noise whilst preserving edges.

As stated in the quantitative evaluation presented above, the most accurate reconstruction is achieved when the scaling is set to a value larger than $s/q_x > 4$. Note that this empirical observation corresponds to the guiding principles based on the theory discussed in Chapter 4.

Thin-Plate regression provides the best accuracy results for small scales $s/q_x < 5$. Since CSRBF and Gaussian require an overlap between RBF centres, the error decreases until over-smoothing occurs. The TV-based regression outperforms other variants when CSRBF is applied with relatively high support ($s/q_x \geq 15$). This is an important finding since it means that using CSRBF at larger scaling values provides comparable approximation quality to *Thin-Plate* kernels while still obtaining a sparse regression matrix K in (5.9). In all observed cases, each of the regularization strategies outperforms the unregularized LSQ accuracy shown in Figure 5.9a. The quality is increased by up to 85% when TVL_1 is applied.



$s/q_x = 2$ $s/q_x = 4$ $s/q_x = 6$ $s/q_x = 10$

Figure 5.10: TVL_1 CSRBF: The effect of support scale.

The findings refer to the simplified case of a 3D step function. However, they can be expected to apply for realistic datasets since the main assumption is that real-world environments consist of many planar connected areas as illustrated by the step function.

5.2.6 Runtime Performance

The final aspect discussed in this section is the runtime performance of the various regression techniques. In particular, since TVL_1 has been selected for its superior shape approximation performance, its computational complexity is higher compared to both TVL_2 or Lasso (Bach et al., 2011). It is thus important to appraise the comparative runtime performance of these regularization methods.

The ADMM algorithm is applied in order to obtain the solution of the TVL_1 optimization task. Its iterative process has previously been discussed in Chapter 4. The first step of this iterative algorithm is to solve a linear system which incurs $\mathcal{O}(N^3)$ complexity. Figure 5.11 shows the time required to solve the TVL_1 task depending on the number of samples N.

It is interesting to note that processing only 2500 samples takes the ADMM algorithm more than a minute to compute. Furthermore, when processing real datasets with more than 100,000 samples, its application becomes prohibitively expensive in terms of computational time. This particular issue will be addressed in Chapter 6, where a novel modification to ADMM is introduced to lower the computational cost by exploiting the sparsity of matrix A in (5.5) when CSRBF is applied.

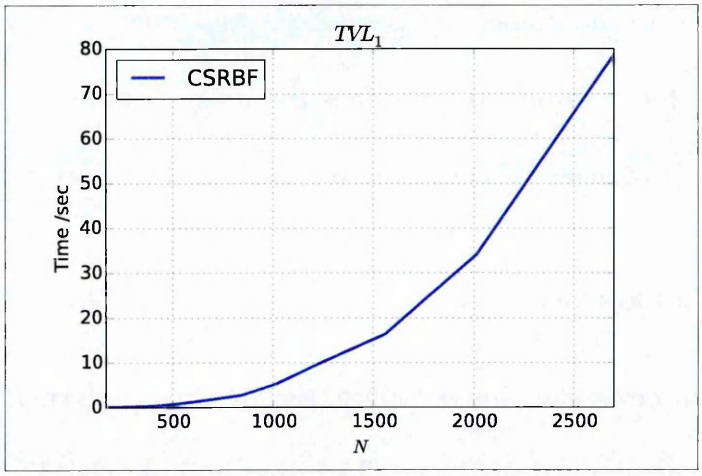


Figure 5.11: The time required to solve TVL_1 via ADMM with N samples.

5.3 Summary

In this chapter it has been shown that when applying TVL_1, regularization with CSRBF significantly increases the 3D shape approximation quality. The integration of the shape variation via $\| D\boldsymbol{\alpha} \|_1$ into the optimization framework enables to identify and to reduce strong shape variation caused by error-prone samples.

So far, the TVL_1 approximation applies the ADMM algorithm as described in Chapter 4. However, since ADMM involves computations of complexity $\mathcal{O}(N^3)$, the technique is not applicable to large point clouds with $N > 10^5$ when N is the amount of processed samples. This issue will be approached in Chapter 6, where a scalable solution is developed to solve the shape approximation task in linear time. Only a linear complexity enables the application of the proposed technique on realistic datasets.

6 TVL_1 for Large Datasets

The previous section presented the novel TVL_1 approach to reconstruct an implicit shape from noisy point clouds. The numerical technique used for optimization, ADMM, suffers from a high order of complexity of $\mathcal{O}(N^3)$, which makes the algorithm infeasible for realistic 3D datasets that comprise several thousands or millions of samples.

In this chapter the CSRBF shape approximation and ADMM are extended, which results in a significant reduction of the complexity to $\mathcal{O}(N)$. This is achieved by applying an iterative numerical solver technique and by avoiding to compute matrices, which further reduces the runtime and storage requirements of the algorithm.

The main contribution here is the application of an iterative numerical solver technique in the second step of ADMM. The minimization of a large linear system is replaced by an iterative method known as Gauss-Seidel (GS) (Saad, 2003). Furthermore, the ADMM-GS implementation does not generate any matrices and works directly on the data, thus no additional memory is required for the numerical optimization to be performed. The result is a fast ADMM method still capable of solving TVL_1 problems in just a fraction of time. The concept, the runtime performance and also convergence properties are presented. The convergence issues are important to consider since an iterative method to solve a linear system is applied.

6.1 Gauss-Seidel for ADMM

Given the Lagrangian from ADMM (4.17), the first optimization step requires the minimization of $L_\rho(\boldsymbol{\alpha}, \mathbf{z}, \mathbf{b})$ with respect to $\boldsymbol{\alpha}$. The minimization is a LSQ task of the general form $E(\boldsymbol{\alpha}) = \| A\boldsymbol{\alpha} - \mathbf{y} \|_2^2$ and can be stated without loss of generality as

$$E(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T A^T A \boldsymbol{\alpha} - 2\mathbf{y}^T A \boldsymbol{\alpha} + \mathbf{y}^T \mathbf{y}. \tag{6.1}$$

The main idea of the Gauss-Seidel approach is to approximate each target variable α_i independently and iteratively by taking its current value as a starting point α_i^0. When expanding the cost term $E(\boldsymbol{\alpha})$ via the Taylor expansion with respect to α_i:

$$E(\boldsymbol{\alpha} + \delta \mathbf{e}_i) = E(\boldsymbol{\alpha}^0) + \delta \nabla_i E(\boldsymbol{\alpha}^0) + \delta^2 H_{ii}$$

with $H = A^T A$ as the Hessian of (6.1), then minimizing for δ gives

$$\delta = -\frac{\nabla_i E(\boldsymbol{\alpha}^0)}{H_{ii}}.$$

When applying this to the least squares regression, the minimum of $E(\boldsymbol{\alpha})$ is approached by

$$\alpha_i^{k+1} = \alpha_i^k + \frac{A_i^T(\mathbf{y} - A\boldsymbol{\alpha}^k)}{A_i^T A_i} \tag{6.2}$$

where A_i is the i^{th} column of A. Note that after α_i was computed, it is updated in place. So the computation of α_{i+1} directly uses the updated α_i.

An important drawback of Gauss-Seidel is its convergence condition. The method requires the maximal eigenvalue of the matrix A^TA to be at most 1 ($\lambda_{max} < 1$). However, the obtained maximal eigenvalue exceeds this restriction when CSRBF is employed. In order to overcome this limitation, *successive over-relaxation* (SOR) is applied, which introduces weighting factor ω to the Gauss-Seidel updates:

$$\alpha_i^{k+1} = \alpha_i^k + \omega \frac{A_i^T(\mathbf{y}_i - A\boldsymbol{\alpha}^k)}{A_i^T A_i}. \tag{6.3}$$

The convergence properties of SOR are discussed in detail by Kahan (1958) and it is shown that ω should be in a range of $0 < \omega < 2$, while $\omega = 2$ maximizes the step size, $\omega = 1$ reduces to the common Gauss-Seidel updates and $\omega < 1$ should only be used when convergence problems arise, which is the case here. The effect of variation in ω will be presented in Section 6.3.

6.2 Matrix-Free Implementation

This section illustrates the computation strategy of the relaxed Gauss-Seidel algorithm from (6.3). The main motivation is to avoid creating matrices before executing an iteration step which reduces the storage requirement and increases the computation speed. Further, updates of α_i only from the neighbouring RBF centres and samples are possible since CSRBFs are employed. The computation speed is significantly increased and the task is solved in only a fraction of time compared to the standard ADMM computation.

Note that the support size and the locations of the RBF centres are estimated

adaptively to the data density as described in Appendix E. Following the concept of the Gauss-Seidel iterations from Section 6.1, the matrix A contains matrices K_n for shape orientation regression, K for zero levels regression and D as a differential part for total variation regularization (see (5.9)). (6.3) can be rewritten as

$$\alpha_i^{k+1} = \alpha_i^k + \omega \frac{y_i - (K_{ni}^T K_n + K_i^T K + \rho D_i^T D)\boldsymbol{\alpha}^k}{K_{ni}^T K_{ni} + K_i^T K_i + \rho D_i^T D_i} \tag{6.4}$$

with $y_i = K_{ni}^T \mathbf{n} + D_i^T(\mathbf{z}\rho - \mathbf{b})$. Considering the case where the matrices K, K_n and D are sparse (see Figure 6.1), the evaluation in (6.4) is simplified by avoiding the multiplication of the zero entries in the matrices. This is achieved by extracting only the neighbouring RBF centres and samples around α_i. The search is performed by an efficient tree search technique (Muja and Lowe, 2014).

The basis functions $\varphi(\mathbf{c}_i, \mathbf{x}_j)$ in

$$K_i^T K = \left(\sum_j^N \varphi(\mathbf{c}_i, \mathbf{x}_j) \cdot \varphi(\mathbf{c}_1, \mathbf{x}_j), \cdots, \sum_j^N \varphi(\mathbf{c}_i, \mathbf{x}_j) \cdot \varphi(\mathbf{c}_M, \mathbf{x}_j) \right)$$

return zero if a sample $\mathbf{x}_j$ is outside of the support of the centre $\mathbf{c}_i$. Thus, all $\varphi(\mathbf{c}_m, \mathbf{x}_j)$ are

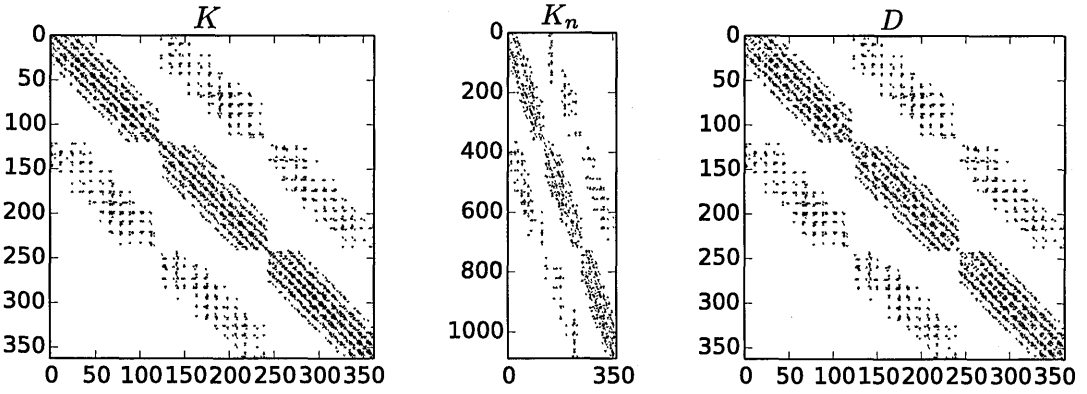


Figure 6.1: Sparse matrices K, K_n and D when CSRBF is applied. Black dots illustrate non-zero entries in the corresponding matrix.

unnecessarily multiplied with zeros. This important observation enables the computation to be reduced dramatically by selecting only the samples around $\mathbf{c}_m$ and neighbouring RBF centres which support intersects with the support of $\mathbf{c}_m$. Repeating this procedure for each RBF centre α_i, the full vector $\boldsymbol{\alpha}^{k+1}$ is calculated.

After $\boldsymbol{\alpha}^{k+1}$ was obtained, steps two and three of the ADMM algorithm follow. The second step involves the minimization with respect to the Lagrangian variable $\mathbf{z}$:

$$\min_z \lambda \parallel z \parallel_1 + \mathbf{b}^T(D\boldsymbol{\alpha} - \mathbf{z}) + \frac{\rho}{2}(\boldsymbol{\alpha}^T D^T D\boldsymbol{\alpha} - 2\mathbf{z}^T D\boldsymbol{\alpha} + \mathbf{z}^T\mathbf{z}).$$

This minimization task can be solved efficiently for each element z_i independently. The solution is known as *shrinkage* and was previously introduced by Tibshirani (1994). It will be described in Appendix B. The last step of ADMM is an evaluation and does not require any minimization techniques. The complete algorithm is summarized in Figure 6.2.

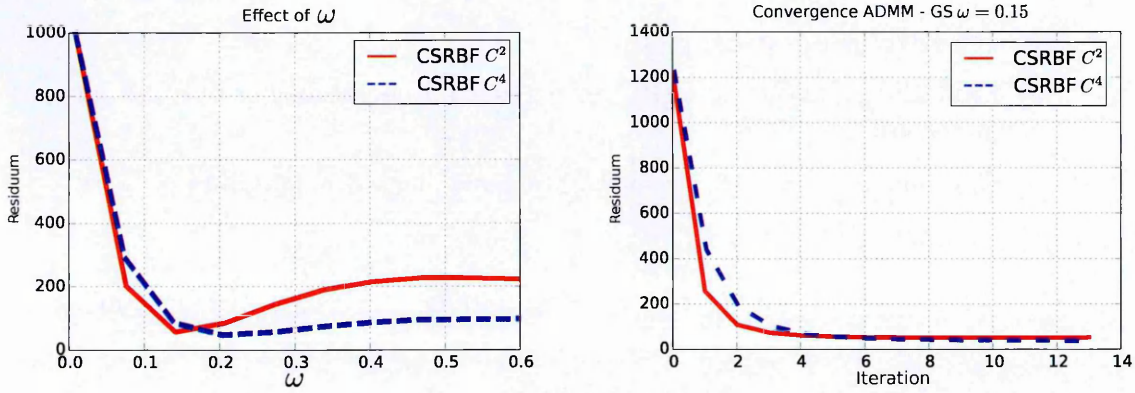
1. For each RBF centre $\mathbf{c}_i$ compute:
   a) Find all neighbouring RBF centres and all neighbouring samples located in the support of $\mathbf{c}_i$.
   b) Compute α_i^{k+1} via (6.4) using only the collected neighbours.
   c) Compute $D\boldsymbol{\alpha}$ with α_i^{k+1}.
2. Solve $\min_z \lambda \parallel z \parallel_1 + \mathbf{b}^T(D\boldsymbol{\alpha} - \mathbf{z}) + \frac{\rho}{2}(\boldsymbol{\alpha}^T D^T D\boldsymbol{\alpha} - 2\mathbf{z}^T D\boldsymbol{\alpha} + \mathbf{z}^T\mathbf{z})$ via *shrinkage* (see Appendix B).
3. Evaluate: $\mathbf{b}^{k+1} = \mathbf{b}^k + (D\boldsymbol{\alpha}^{k+1} - \mathbf{z}^{k+1})\rho$.

Figure 6.2: Fast matrix-free TVL_1 algorithm.

6.3 Convergence and Runtime Performance

The previous chapters presented the extension of the ADMM approach with Gauss-Seidel and the weighting parameter ω leading to a SOR method. Since iterative optimization techniques approach a solution via multiple steps, it is important to address the convergence speed. If an algorithm approaches an optimal solution very slowly, many iterations are required, which increases the overall runtime of the method. The introduced parameter ω directly controls the step width of a single iteration. However, a high ω value (≈ 2) breaks the optimization and the solution diverges from the optimum. This section evaluates the effect of ω on CSRBF-C^2 and C^4 RBF and a ω value is proposed for both RBF types.

The introduction of SOR with the iteration step width ω is necessary since the largest eigenvalue of the matrix A^TA in (6.1) does not meet the convergence condition for Gauss-Seidel (Kahan, 1958). Moreover, since no theoretical estimation of the step size parameter ω is possible, its value is estimated empirically. Figure 6.3a shows the effect of ω on the minimization process when CSRBF-C^2 and CSRBF-C^4 are applied after



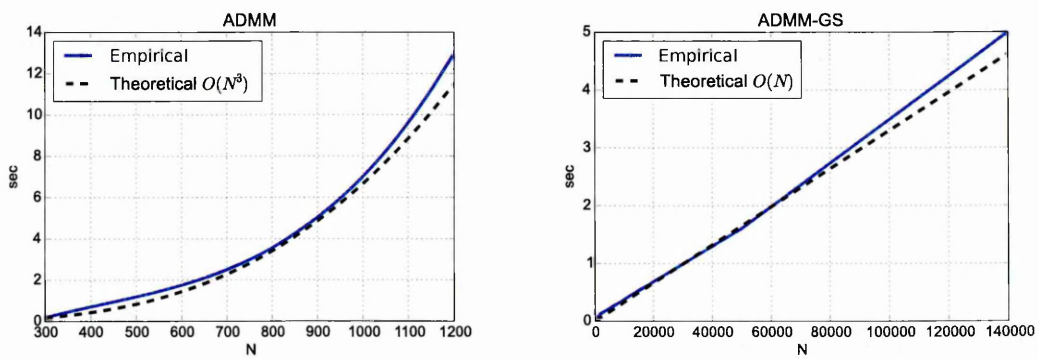
a) b)

Figure 6.3: a) The effect of the SOR weighting ω. b) The convergence performance for selected ω for CSRBF-C^2 and CSRBF-C^4.

five iterations. The graph shows the residuum, namely the achieved cost value with the estimated $\boldsymbol{\alpha}$ of the optimization after four iterations. Figure 6.3b shows the convergence performance when $\omega = 0.15$ is selected. Compared to CSRBF-C^4, CSRBF-C^2 converges faster because of its higher stability properties.

The extended ADMM-GS approach enables the TVL_1 minimization to be solved much faster than the standard technique presented in Chapter 5. Figure 6.4 shows the achieved runtime performance processing the synthetic 3D step function increasing the number of samples. For the experiments, the size of RBF was set to $s/q_x = 5$. Larger values generally reduce the sparsity of the matrix K and increase the computation time.

The proposed ADMM-GS technique clearly outperforms the standard ADMM. Additionally, the verification of the new low complexity $\mathcal{O}(N)$ is presented. The computation time achieved by the algorithm is compared to the theoretical linear runtime. The computation time is generally slightly higher than the theoretical prediction since memory allocation and thread blocking by the operating system causes additional delays (Kopetz, 2011). Intel i5 with 4GB of memory has been used to compute the runtime performance. More advanced computing platforms will enable shorter processing times.



a) b)

Figure 6.4: Runtime complexity validation. a) ADMM, b) ADMM-GS.

6.4 Summary

This chapter presented an extension to the ADMM optimization technique that allows to solve the TVL_1 task for large datasets. The new computation strategy exploits the sparsity due to the compact support of CSRBF. Further, a matrix-free optimization procedure is established which works directly on the input points instead of allocating additional memory for linear algebra computations. Since all multiplications with zeros are avoided in the optimization process, the complexity is reduced significantly to $\mathcal{O}(N)$. As presented in Figure 6.4, the proposed technique is 100 times faster when $N = 1200$ samples are processed to a 3D shape. The standard ADMM is not applicable for larger datasets with $120,000$ samples. A drawback of the presented methodology is the high number of iterations required to solve the framed problem. Since the SOR weighting ω is introduced to relax the convergence condition imposed by the Gauss-Seidel algorithms, the number of iterations may increase depending upon the dataset being used.

Chapter 7 will focus on the qualitative and quantitative evaluation of the proposed technique. The algorithm will be evaluated on realistic data with available ground truth and challenging datasets obtained from a mobile stereo camera system.

7 Performance Evaluation

This chapter compares the proposed TVL_1 shape reconstruction model with two state-of-the-art surface approximation techniques (Section 7.1) which are applied to the three test datasets presented in Section 3.7. The critical evaluation consists of two distinct elements: a qualitative assessment (Section 7.2) which focuses specifically on the visual appearance of the reconstructed shapes; and a quantitative assessment (Section 7.3) which applies the quality metrics introduced in Chapter 3. Section 7.4 will conclude this chapter by highlighting the generic findings common for all datasets when TVL_1 is applied.

7.1 Surface Reconstruction Comparators

The proposed TVL_1 approach is compared with two existing surface reconstruction techniques: the Poisson algorithm (Kazhdan and Hoppe, 2013) and SSDF (Calakli and Taubin, 2011). Since the Poisson algorithm is preferred in many application domains (Rusu and Cousins, 2011), it is a comparative technique of particular interest to this research. Similarly, SSDF was one of the first algorithms to integrate the smoothness regularization into the shape reconstruction process, which equally makes this a valuable comparator in the critical evaluation.

Both methods differ from the proposed TVL_1 approach in several important aspects:

- **User-specified subdivision depth:** In contrast to the comparators, the proposed framework adapts the RBF sizes to the data. Poisson and SSDF group the 3D samples via an OcTree (Gomes et al., 2009) into disjunct cubes and the depth of the tree is set by the user before the approximation process starts. Unfortunately, the depth parameter varies depending on the dataset volume, which makes the other compared methods impractical for non-interactive shape reconstruction. In both cases, the user has to estimate the proper depth before applying one of the comparator algorithms. More details about the RBF distribution can be found in Appendix E. First of all, the evaluation presented in Appendix D has to be performed in order to estimate the optimal dataset-specific depth parameter for Poisson and SSDF.

- **Discretization:** Poisson and SSDF work explicitly with the discretized OcTree data structure. Both methods require that a surface between neighbouring OcTree leaves has to exist, which can lead to the reconstruction of non-existent surfaces. While this effect can be beneficial for hole-filling by extrapolation, it also causes the closure of larger naturally occurring apertures like doors, windows and occluded areas. This is disadvantageous in many applications involving both inspection and exploration.

- **Minimized cost function:** The Poisson approach solves the partial differential equation $\triangle f = \nabla \mathbf{n}$ by performing multiple convolution (weighted averaging) steps that lead to smooth results (Kazhdan et al., 2006). Instead of modelling the

function f by RBF, constant basis functions, $\varphi = 1$ centred on the neighbouring OcTree leaf corners are applied. Subsequently the solution is approached by a coarse-to-fine optimization approach exploiting the hierarchical structure of the OcTree. The SSDF approach achieves smoothness regularization by applying discrete derivatives and the standard L_2 norm. Using this particular norm rather than L_1 does not take into account the sparse number of edges in a 3D scene. This leads to an overall smoothing of the shape reconstruction results, as expected from the L_2 regularization discussed in Chapter 4.

The first comparative performance evaluation will be considered now.

7.2 Qualitative Analysis

In this section, visual assessment of the 3D shape reconstruction results will be undertaken. Section 7.2.1 will present the reconstructed meshes for the *Facade* and the *Kitchen* datasets while Section 7.2.2 will examine the related approximation errors in the reconstructed shapes.

7.2.1 Shapes from Synthetic Data

Figures 7.1 and 7.2 show the reconstructed surfaces for the *Facade* and the *Kitchen* datasets using the Poisson, SSDF, TVL_1-C^2 and TVL_1-C^4 techniques respectively.

In addition to the full 3D datasets shown in Figures 7.1a and 7.2a, two particular areas have been enlarged to highlight the quality of the respective shape reconstructions in especially challenging areas (sparse samples, occlusions, fine structures) as shown in Figures 7.1c and 7.2c. The corresponding ground truth models are also included for

subjective assessment purposes.

A general perceptual conclusion for both of these synthetic datasets is that both proposed TVL_1 regularization techniques consistently achieve superior reconstruction, particularly in terms of the detail in the vicinity of and around corners and edges (see b, c in both Figures 7.1 and 7.2). Note that since the balcony railings in the *Facade* dataset are represented by a pure texture, its shape is not being sampled by the simulated camera and thus cannot be reconstructed by any of the shape modelling techniques.

The reconstruction along unsampled areas such as the roof of the *Facade* incurs holes since TVL_1 avoids extrapolation of the samples. Similar behaviour can be seen in the *Kitchen* dataset along the boundary edge between the floor and the furniture (Figure 7.2b).

When processing sparse samples collected from either a narrow or reflective structure such as in Figure 7.2c, TVL_1 is the only technique that successfully identifies this as a surface. For example, neither Poisson nor SSDF considered the 3D samples around the water tap and so failed to reconstruct any shape at all. However, the surface orientation quality suffers from a limited number of normals, because its computation relies on a high density of 3D samples (Rusu and Cousins, 2011) which the test datasets do not possess.

7.2.2 Region-Specific Error Distribution

Seven challenging areas for shape reconstruction have been identified in each dataset to assist in the critical evaluation of the 3D models. In evaluating the capabilities of the different algorithms, a number of feature areas have been identified, namely:

areas with a low sampling density (①,②,④) including occlusions (④, ⑥) and fine structures (③, ⑤). The comparative performance of each of the methods is illustrated

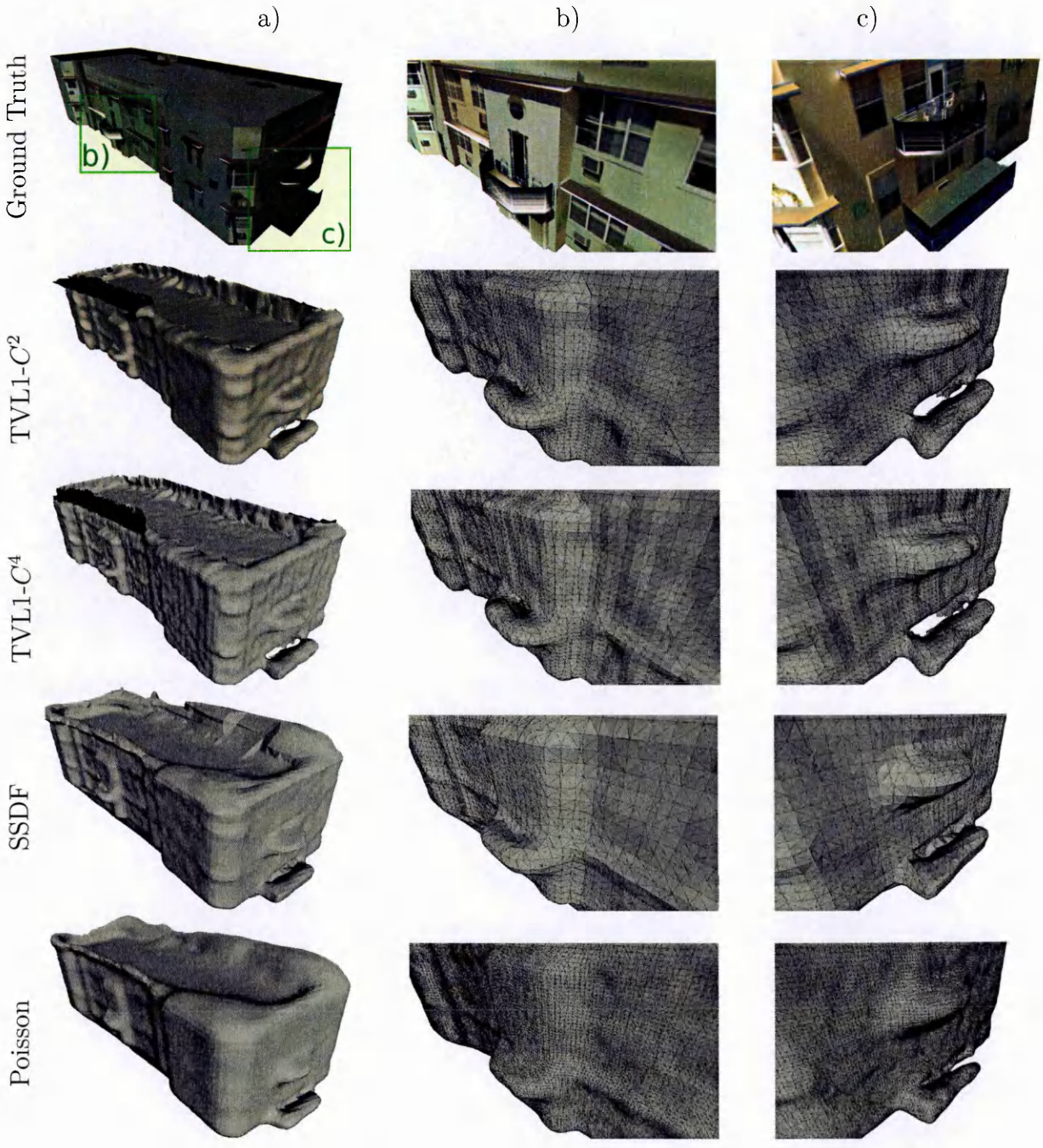


Figure 7.1: Reconstruction results on the *Facade* dataset. a) Full model overview, b) front view enlarged, c) side balcony enlarged.

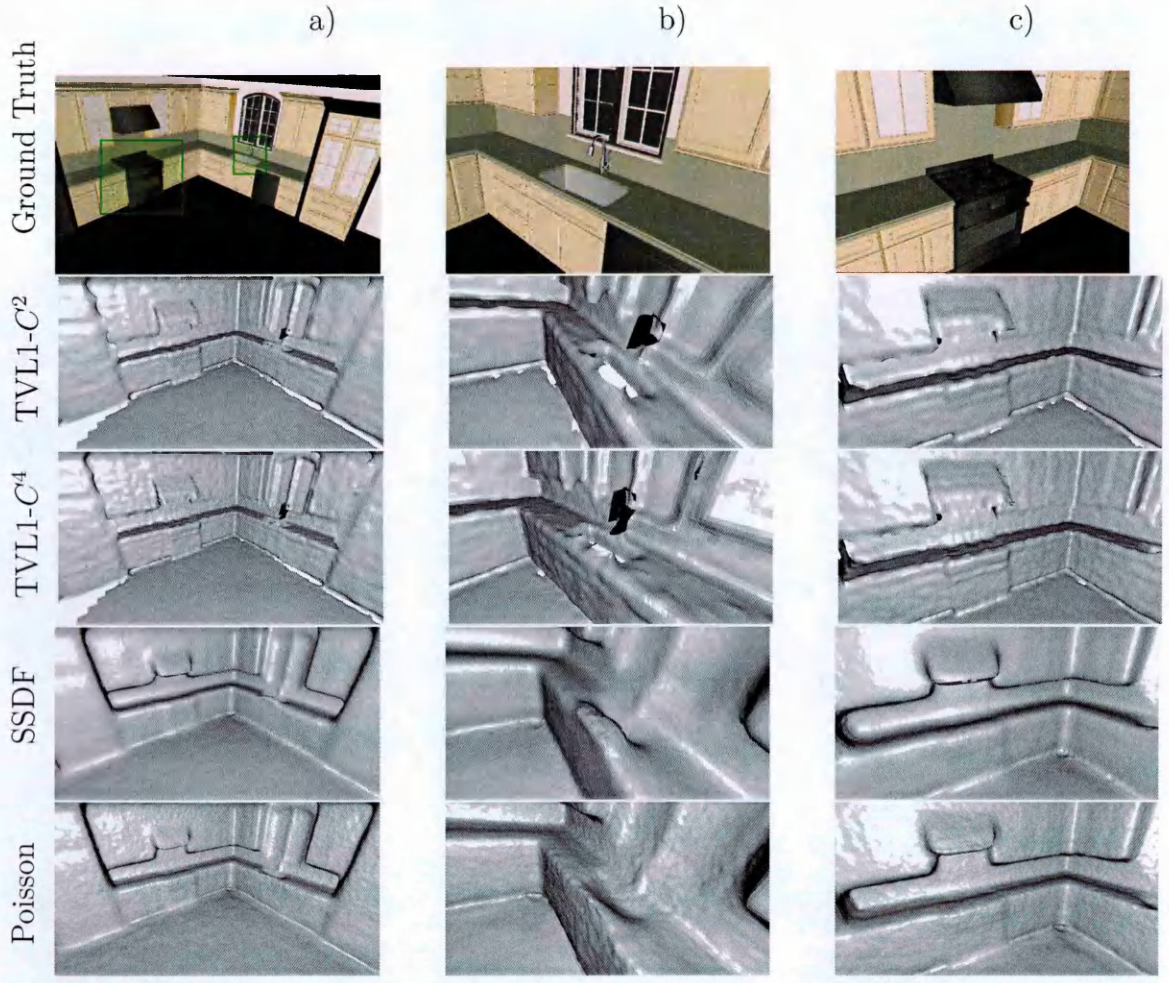


Figure 7.2: The reconstruction results of the *Kitchen* dataset. a) Overview, b) water tap, c) stove.

in Figure 7.3, where the spatial distribution of the point-to-point errors aligned to the ground truth model is coloured in red. This enables areas that are difficult to be reconstructed for a particular technique to be highlighted. The ground truth model and the reconstructed mesh are compared with the distance between each ground truth sample and its nearest vertex in the reconstructed model being determined. The error distributions are shown by the red colours in Figure 7.3, where the brighter the colour, the higher the error. In the first row a series of circled numbers have been identified in the error distributions to aid the critical discussion of the respective models. These will

now be discussed individually referring to Figure 7.3:

1. In area (1), where two planes interconnect and rectangular corners need to be reconstructed, TVL_1 achieves higher accuracy compared to both Poisson and SSDF, which apply greater smoothing to the approximation and as a consequence generate notably larger errors. This can be seen on the example area beneath the exhaust hood in the *Kitchen* dataset.

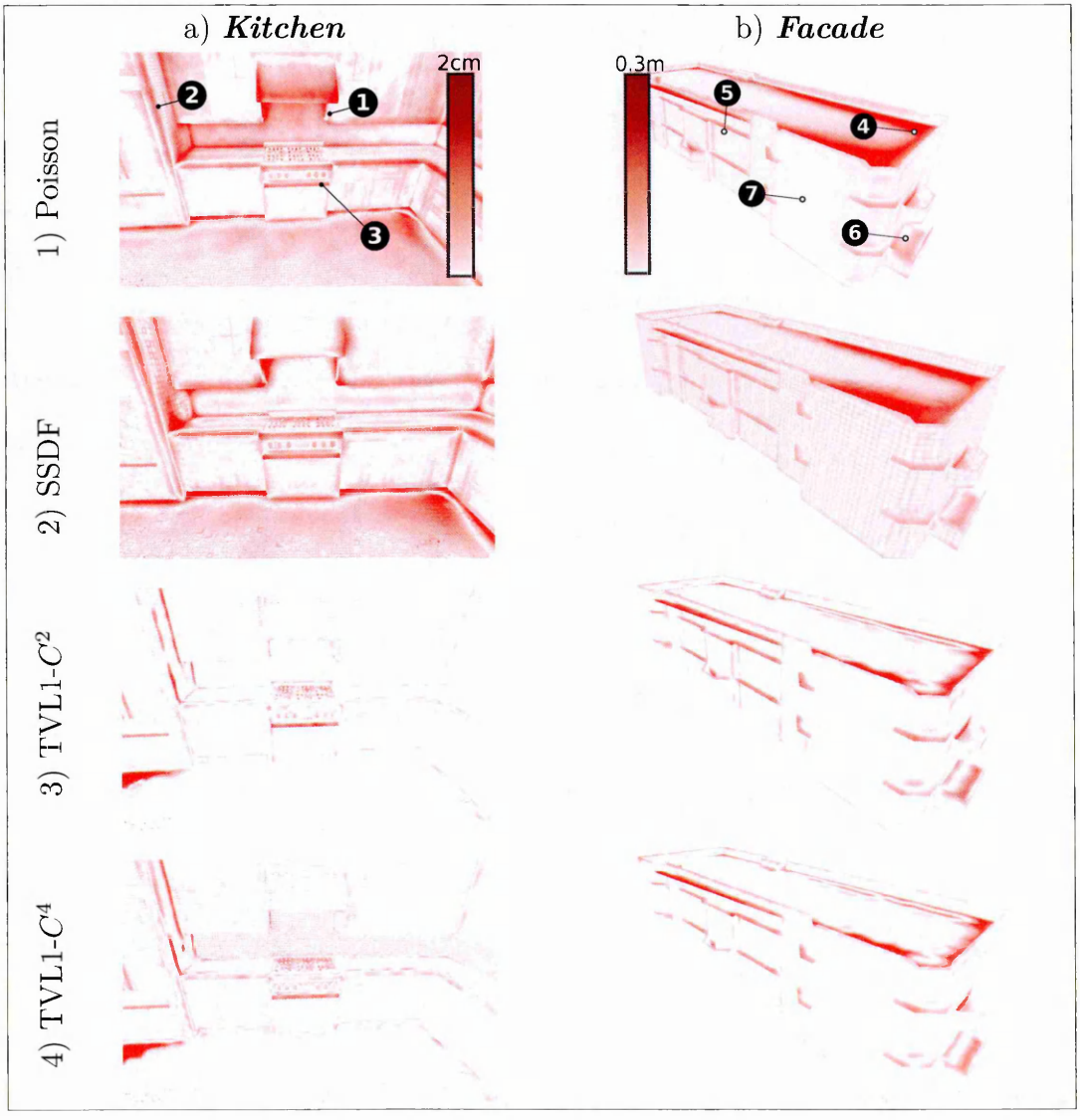


Figure 7.3: Error distribution for the *Kitchen* (a) and the *Facade* (b) datasets. Red colour illustrates larger errors.

2. When the noise variance is small with respect to observed structures in a scene (e.g. area (5)), all the analysed reconstruction methods are able to approximate shape geometry effectively, though at different quality levels. However, when the noise variance exceeds the size of the objects (e.g. the fine joints in the *Facade* dataset in Figure 7.1) then these become smoothed and disappear in the reconstructed shape approximation. Note that also the planarity integration regularization can not change this principle of too low *signal to noise ratio.*

3. In large planar areas such as in (7) the reconstruction quality of TVL_1 is consistently superior to Poisson and SSDF. This can be observed along the wall of the *Facade.*

4. In situations where no 3D samples are available such as in the corner of the roof in the *Facade* both Poisson and SSDF still extrapolate the holes into a closed surface and thus create an erroneous reconstruction. In contrast, TVL_1, as seen also in (2), does not perform any hole-filling, i.e. it skips this area.

5. In regions of a scene which are weakly sampled and contain sharp corners (see (2)) both SSDF and Poisson force the surface to close between samples, which leads to large errors as highlighted in Figure 7.3b. This is not advantageous for any dataset which contains naturally occurring apertures like open doors or windows. In contrast, for TVL_1 the shape reconstruction of large unsampled areas is not performed since the error in such regions would be too large and would not reflect the true geometry.

6. When parts of an object are occluded such as in the case of the floor of the balcony

in the *Facade* dataset, the reconstructed shape is successfully extrapolated by all methods, though focusing on samples rather than closed surfaces. That means the results for both SSDF and Poisson lead to larger errors than either TVL_1-C^2 or TVL_1-C^4.

7. While fine structures (3) tend to be smoothed by all approximating techniques (e.g. stove in the *Kitchen* dataset), the new TVL_1 method as illustrated in Figure 7.1 is still able to extract rudimentary shapes like the cabinet handle that are not visible in the corresponding results for both SSDF and Poisson. Furthermore, in comparison to TVL_1-C^2, TVL_1-C^4 provides a consistently better approximation of finer details in the reconstruction. The underlying reason for this is that structures which exhibit significant shape variations benefit from the higher-order interpolation in the shape approximating process.

A number of general observations concerning the performance of the new TVL_1 technique can be distilled from this qualitative evaluation:

- TVL_1 focuses on the approximation of samples without enforcing and synthesizing closed surfaces.
- TVL_1 is more robust in being able to deal with larger noise variance than either Poisson or SSDF. However, when the noise variance exceeds the size of the sampled object, only a vague smoothed reconstruction of the object is possible.
- TVL_1 leads to significantly lower over-smoothing effects.
- TVL_1 clearly outperforms both Poisson and SSDF when large planar areas need to be reconstructed from relatively dense samples.

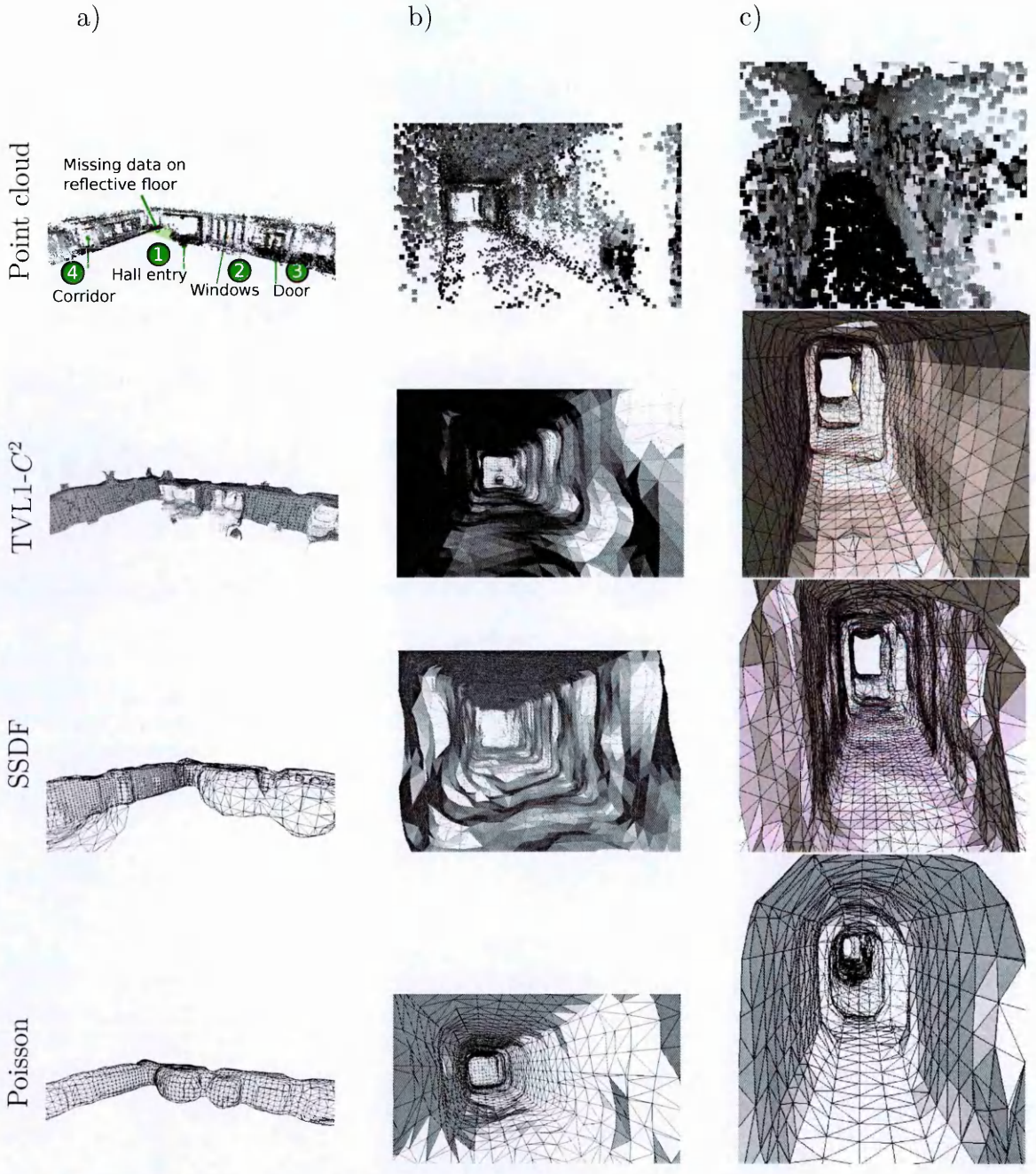


Figure 7.4: The reconstruction results of the *Corridor* dataset. a) Corridor crossing, b) reflective floor (2), c) corridor enlarged (4).

7.2.3 Shapes from a Non-Synthetic Dataset

Unlike the two synthetic datasets *Kitchen* and *Facade*, this 3D dataset comprises a physically acquired point cloud from a stereo camera as shown in Figure 7.4. The reason why in this case no detailed quantitative analysis can be performed is the lack of ground

truth. The entrance area (1), windows (2) and doors (3) do not naturally contain samples and thus should not become enclosed by surfaces. However, in both SSDF and Poisson approximations such apertures become closed surfaces and consequently create non-existing walls in the shape reconstruction as illustrated in Figure 7.4a.

In Figure 7.4b a scene with reflective floor and glass windows along the corridor has been observed. This represents an extremely challenging task since reflective objects are generally difficult to handle for optical sensors. Provided the holes in the dataset are not larger than the sampling density, then the adaptive RBF distribution enables a satisfactory approximation of the surface to be made. In comparison to SSDF and Poisson, the resulting shape approximation has fewer ripples and does not suffer strong over-smoothing effects as in the Poisson case. Furthermore, while SSDF reconstructs more structural details than Poisson, both are more sensitive to noise than the proposed TVL_1 approach.

7.3 Quantitative Analysis

In this section, a quantitative analysis is performed to assess the various shape reconstruction methods using different evaluation metrics. Since the ground truth data is available, the quality metrics introduced in Chapter 3 are applied to measure the deviation of the approximation results from the reference models.

The histograms in Figures 7.5a and b show the error density distribution with respect to the distance from the reference shape. This basically describes how many samples are a distance away i.e., $0.01m$, from the ground truth surface. The smaller the overall distance from the ground truth shape, the higher the peak and the closer it appears

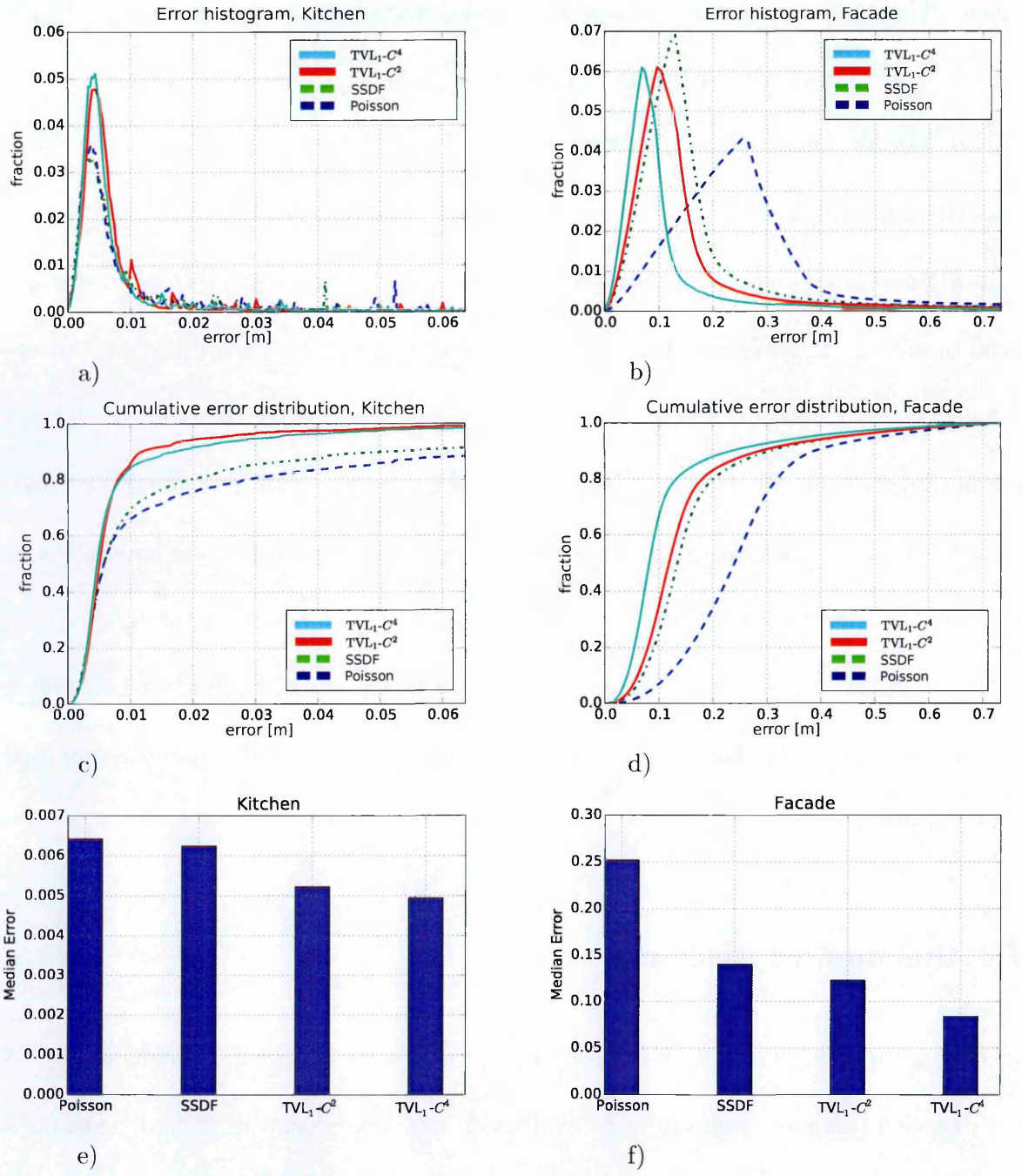


Figure 7.5: Statistical quality metrics: a, b) error histograms, c, d) cumulative error distributions, e, f) median errors in metres.

to the origin of the graph. In the case of the *Kitchen* dataset, the peak is much more to the left (5mm) since the density of the samples is much higher than in the *Facade* dataset (0.1m).

In both scenarios, *Kitchen* and *Facade*, TVL_1 outperforms Poisson and SSDF in

terms of their corresponding shape reconstruction accuracy. This is especially true for the *Facade* dataset, where TVL_1 distributions produce consistently lower error than its comparators, especially the Poisson approximation. The reason for the TVL_1 improvement is that the noise in the plane object samples is better suppressed due to the integration of *a priori* knowledge that a planar surface is approximated.

The cumulative distribution graphs in Figures 7.5c and d reveal how many samples have a smaller error than say $0.2m$. These clarify that for both TVL_1 variants 90% of the errors are less than 1.5cm for the *Kitchen* and 0.2m for the *Facade* dataset respectively. In contrast, the corresponding errors for Poisson are 5cm and 0.4m, and for SSDF they are 4.5cm and 0.33m.

Figures 7.5e and f illustrate the respective median errors, where the TVL_1 variants again achieve the smallest values. For the *Kitchen* dataset, noise is relatively low and the data density high, so all the shape reconstruction methods produce satisfactory results. However, for the *Facade* dataset, where the noise level is high, the TVL_1 framework significantly outperforms its comparators. Overall an improvement of up to 23% on the *Kitchen* and up to 67% (compared with Poisson) on the *Facade* dataset has been achieved corroborating the corresponding perceptual results for these datasets.

Referring to the noise level applied to the ground truth data of $\sigma = 0.4$m for the *Facade* and $\sigma = 0.05$m for the *Kitchen* dataset, the resulting TVL_1 standard deviations are significantly lower. TVL_1 reduces the effect of the noise by up to 42% (*Facade*) and 26% (*Kitchen*) as shown in Table 7.1. The table also presents the input noise for both datasets (*Facade* = 0.4m and *Kitchen*: = 0.05m). TVL_1 achieves to suppress the noise down to 42% (*Facade*) and 26% (*Kitchen*) as shown in Table 7.1. The table presents

the input noise for both datasets (*Facade*: $\sigma = 0.4$m and *Kitchen*: $\sigma = 0.05$m) and the corresponding variance of the reconstructed shape. The reconstruction variance has been computed using the ground truth data and numerically measures the shape approximation accuracy. This is particularly interesting since noise in 3D samples makes the shape reconstruction challenging. Yet despite this, the resulting errors are reduced by up to 80% when TVL_1 is applied.

Table 7.1: Error Statistics. Median (Med.), Variance (Var.) and $\sigma_{rel} = \frac{\sigma_{output}}{\sigma_{input}}$

	Facade, $\sigma = 0.4$m			Kitchen, $\sigma = 0.05$m		
	Med.	Var.	σ_{rel}	Med. (10^{-3})	Var.	σ_{rel}
Poisson	0.25m	0.35m	87.5%	6.4m	0.17m	300%
SSDF	0.14m	0.33m	82.5%	6.2m	0.08m	160%
TVL_1-C^2	0.12m	0.24m	60%	5.2m	0.010m	**20%**
TVL_1-C^4	0.08m	0.17m	**42.5%**	4.9m	0.013m	26%

While these findings have focused on two very different test 3D datasets, the findings relating to the comparative performance of TVL_1 can be generalized and the dataset characteristics under which superior shape reconstruction results can be achieved will now be summarised: In critically evaluating the performance of TVL_1 in comparison with existing state-of-the-art 3D shape reconstruction techniques, it performs better when:

- Approximating details at corners and along contours by virtue of *a priori* planarity knowledge are incorporated into the shape reconstruction process.
- By distributing the RBF centres adaptively to the dataset, TVL_1 does not reconstruct non-existent surfaces in larger holes like doors or windows. This is essential for generic applications in many domains.

- Processing noisy datasets, a suppression of up to 50% can be achieved.

7.4 Summary

This chapter presented a thorough evaluation of the proposed shape approximation using TVL_1 regularization. Two state-of-the-art comparative methods, Poisson and SSDF, were applied to three diverse 3D test datasets to assess the quality of their respective shape approximations. The datasets were *Facade*, *Kitchen* and the physically-acquired point cloud *Corridor*, which each exhibit different and challenging features, sampling densities and noise levels. The critical evaluation involved a visual assessment of the 3D reconstructed shapes supported by a numerical comparison of the approximation methods. Findings from the evaluations reveal conclusively that the new TVL_1 technique produced higher accuracy 3D shape approximations in all test scenarios.

8 Future Directions

This thesis has presented a novel robust 3D shape approximation framework with significant improvements being achieved over existing state-of-the-art techniques due to the integration of *a priori* smoothness knowledge via TVL_1 regularization. The framework is flexible and adaptive to data and does not require user interaction on specific datasets. The following sections outline some suggestions for extending the framework in order to pave the way for future research.

8.1 Recursive Shape Approximation

The underlying thesis focuses on the shape reconstruction considering all available 3D samples in a single optimization process. In entertainment, tele-operation, virtual reality or robotic applications it is of high importance to reconstruct the scene from a data stream (e.g. a video of depth images). This means that the shape approximation process is expected to use new incoming data packets to update and improve the existing 3D shape iteratively. This is certainly a feasible extension to the framework since the TVL_1 regularization technique already involves iterative optimization steps that can further be amended to update the shape from incoming measurements. The recursive extension of the presented framework will enable the reconstruction of dynamic environments and

also the application of consumer grade 3D perception hardware like small cameras or scanners which generate error-prone data.

8.2 Data Structures for Large Environments

3D datasets of few million 3D samples have been processed so far. However, in real-world environments which may cover hundreds of thousands of kilometres the requirement will involve processing billions of data points. Unfortunately, state-of-the-art data structures are only able to access distances of a few hundred metres and are not scalable to large domains. For this reason, effective data storage techniques need to be developed enabling fast access to 3D data that may be simple 3D points or preprocessed implicit shapes. This extension of the TVL_1 framework is expected to have an additional impact on a range of different industries such as *geographic information systems* (GIS), autonomous driving in public or in industrial facilities, and security applications that monitor larger environments such as construction sites below or above the ground.

8.3 Automated Scene Understanding

After having reconstructed the implicit shape model, the highest value from the reconstructed shapes is expected when object detection or general scene analysis is performed automatically. Dealing with the implicit shape representation, a scene can be analysed by its differential patterns such as the gradient and second derivative, which can be viewed as the planarity of a shape. Differential analysis is the fundamental baseline of many convolutional *Deep Learning* techniques (Lai, 2015; Goodfellow et al., 2015; Yan et al., 2014), which can directly be applied on top of the reconstructed implicit

shape. Currently, *Deep Learning* outperforms many other classification methods, which leads to the overall conclusion that integrating *Deep Learning* into the newly developed framework will enable a powerful system understanding of the surrounding environment. This opens up new opportunities in areas such as the detection of pedestrians, cars and other objects, which is pivotal in many automation domains.

8.4 Out-of-Core Rendering

When processing huge 3D sample datasets, the visualization of a virtual environment becomes very challenging. Since the datasets cannot be stored in the internal memory of a GPU, direct visualization of the shapes or raw 3D samples is not possible even on high-end hardware. This requires the development of new multi-level memory management techniques. Depending on the camera position, only the observed parts of a 3D scene are loaded into the internal GPU memory while the rest of the dataset is ignored. This issue has already been recognized by numerous research groups and several approaches have been proposed (Gobbetti et al., 2008) to deal with large point clouds or meshes.

Recent research by Limper et al. (2014) and Robinet et al. (2014) has focused on 3D visualization within web browsers. There the term *core* defines a local machine that downloads only the relevant part of a 3D scene from a web server. In combination with the proposed recursive shape approximation strategy discussed in Section 8.1, this opens many new perspectives on how 3D content can be generated in real time for human-to-machine interactions via the mobile internet.

9 Conclusion

The increasing demand for efficient automation in many industries has led to extensive research into mobile robotics. The demand is driven by several domains including automated transport, agriculture machinery and inspection applications. A fundamental requirement in these applications is to be able to analyse the environment in which the robot operates in real-time by using either laser scanners and/or cameras. Such devices, however, only provide unordered 3D samples. A major challenge is that these samples do not provide any intrinsic information about the scene and thus have to be processed to derive more meaningful structures like surfaces, apertures and obstacles which then enable a robot to navigate.

This thesis has presented a new framework which addresses the fundamental question of 3D shape reconstruction from incomplete 3D samples by the integration of *a priori* smoothness knowledge. In modelling outdoor and indoor scenes, it is intuitive to assume most observed surfaces are planar. This *a priori* knowledge has been formalized by means of the convex regularization and extended to a robust model using the L_1 norm. This integration of planarity information into the 3D shape approximation process significantly increases the robustness of the framework to measurement errors. The new knowledge integration framework presented in this thesis makes three original scientific contributions to the 3D computer vision and robotics field:

1. The most significant contribution is the new planarity-aware 3D shape approximation algorithm that integrates *a priori* knowledge as piecewise smoothness with the inducing TVL_1 regularization enabling more robust shape reconstruction from error-prone 3D samples.

2. A novel formalization of implicit shape representations using RBF has been developed. This enables the representation of arbitrary non-linear shapes by means of linear functions and thus facilitates TVL_1-based shape approximation from scattered 3D samples.

3. A novel numerical technique has been developed to reduce the order of computational complexity of the shape reconstruction process from $\mathcal{O}(N^3)$ to $\mathcal{O}(N)$ without compromising the accuracy, noise suppression performance or the approximation of piecewise smooth shapes. This importantly allows the new framework to be applied to realistic 3D datasets which typically comprise millions of data samples.

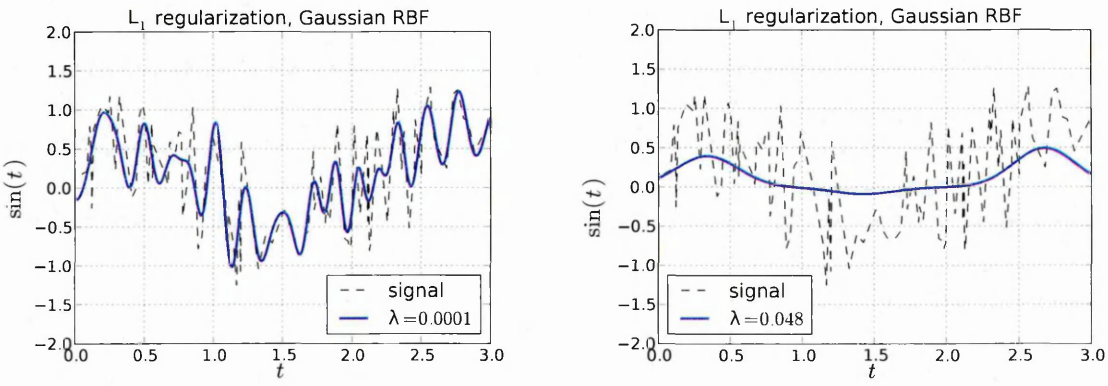
The contributions of this thesis have an impact on many different robotics domains where spatial perception is required. The main benefit of the presented research is the accuracy of the reconstructed shape even if error-prone 3D points are provided. This is of particular interest for low-cost 3D sensors including consumer grade cameras or simple 3D scanners which are usually employed in lightweight systems. Applications such as inspection via robots or drones, 3D modelling with low-cost consumer hardware, or navigation and path planning in unknown environments are only a few examples where the presented research has an impact. Critical tasks such as obstacle avoidance or human-machine-interaction naturally rely on accurate 3D modelling, which makes the presented research contributions significant for the considered technologies.

Appendix A: Estimating the Weighting Parameter λ

Background

Chapter 4 showed how the L_1 regularization can solve the stability and the over-fitting issues. However, while applying the L_1 regularization, it is important to select a proper weight λ to the penalty term in equation (4.15). Setting λ to a very high level leads to over-smoothing. A weight that is too small leads to over-fitting and instable approximation.

Consider Figure A.1, in which two approximation results with two different weights are demonstrated. Since the residuums of the minimized cost function do not provide information about over-fitting or over-smoothing, a technique assessing the overall approximation quality is required. In the field of machine learning, this issue is usually



a) b)

Figure A.1: The effect of the smoothing parameter λ on the regression of a corrupted signal. a) λ is too small, b) λ is too large.

approached by splitting a dataset into a training and a testing set. The training set is used for the estimation of the target variables $\boldsymbol{\alpha}$ while the testing set is applied to measure the quality of the model defined by $\boldsymbol{\alpha}$. However, applying this technique to robotic applications is usually not possible since no separated data samples are available.

Because of these restrictions, Mosteller and Wallace (1963) and later Allen (1974) proposed a method known as the *cross validation* (CV) or *leaving one out* technique. Basically, it is assumed that the best value for λ lies in a range between λ_{min} and λ_{max}. Dividing this range into M discrete λ_m values enables the best of them to be evaluated and selected. In order to estimate the quality of each λ_m, a single sample from the dataset is omitted, the regression task is solved and the error of the omitted sample is measured. This is repeated for each of the N samples.

The full search would require performing the regression $\mathcal{O}(M_\lambda \cdot (N-1))$ times when evaluating all M hypotheses for the weight λ. Note that when the computation of $\boldsymbol{\alpha}$ is of complexity $\mathcal{O}(N^3)$, the cross validation will have $\mathcal{O}(N^4)$. In practise this can be performed for very small datasets of $N < 1000$ samples.

Bach (2008) proposed selecting only $k < N$ samples randomly and to extract the most influential coefficients in $\boldsymbol{\alpha}$. Meinshausen and Bühlmann (2010) defined the importance of the coefficients via probability to be part of the solution. The detection process is performed for every λ hypothesis, so that the overall regression task reduced to $\mathcal{O}(M_\lambda \cdot k)$ computations. Unfortunately, the number k has to be high in order to obtain reliable estimates. Thus the repetitive solving of the regression task still causes high computational costs.

Following the ideas for L_2 regularization from Wahba (1990), a more efficient method is applied. The approach does not require solving the regression task for each of k^{th} samples. The regression task is solved only once for each hypothesis λ_m, thus reducing the complexity to only $\mathcal{O}(M_\lambda)$. The technique is adopted here to L_1 regularization and will be described in the following paragraph.

Generalized Cross Validation

In ordinary CV the parameter $\lambda = \lambda_m$ is fixed in the first step, one of the samples $k \in \{1, \cdots N\}$ is excluded, and the regression task is solved. Finally, the overall error of the interpolated model with omitted sample k is estimated. This process is repeated for all given samples N. The variance of all N errors for λ_m is the metric:

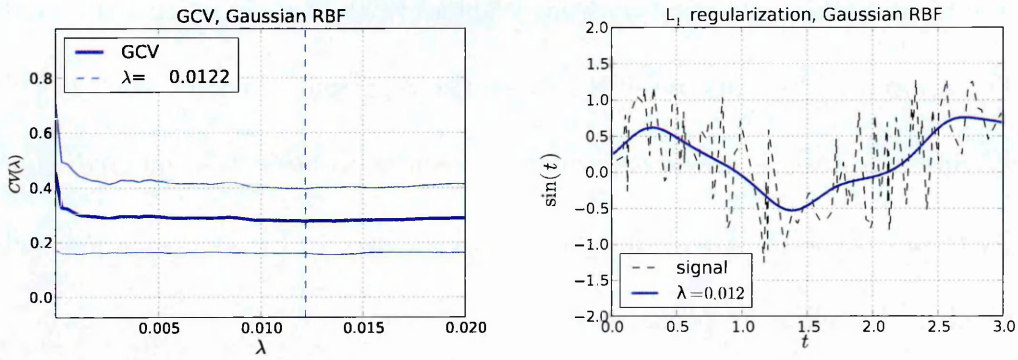
$$CV(\lambda) = \frac{1}{N}\sum_{k=1}^{N}(y_k - f_\lambda^{\not k}(x_k))^2$$

where $f^{\not k}$ is the estimate of f without k^{th} sample. So the best λ^{opt} is the minimizer of $CV(\lambda)$ which is found by calculating $CV(\lambda)$ for all $\lambda_m, m \in \{0, \cdots, M_\lambda\}$. When considering the L_2 regularization norm as studied by Wahba (1990), the estimator of f including all samples is

$$f_\lambda(x) = k^T\alpha^{opt} = k^T(K^TK + \lambda D^TD)^{-1}K^Ty.$$

Then the $CV(\lambda)$ error follows the identity:

$$\frac{1}{N}\sum_{i=1}^{N}(y_i - f_\lambda^{\not i}(x_i))^2 = \frac{1}{N}\sum_{i=1}^{N}\left[\frac{y_i - f_\lambda(x_i)}{1 - S_{ii}}\right]^2$$



a) b)

Figure A.2: a) $CV(\lambda)$, b) approximation result with optimal λ^{opt}.

and in matrix form:

$$\equiv \frac{1}{n} \parallel (I - S(\lambda))y \parallel^2 / \left[\frac{1}{n}\mathrm{Tr}(I - S(\lambda))\right]^2 \tag{A.1}$$

with $S = K(K^TK + \lambda D^TD)^{-1}K^T, S \in \mathbb{R}^{N\times N}$. This identity states that the $CV(\lambda)$ may be computed from only a single estimation of f_λ without having to repeat the computation N times as it is the case in standard methods. Now, this CV approach is extended for L_1 regularization.

Figure A.2 illustrates the estimated λ for the problem of reconstructing a sin signal via Gaussian RBF ($s = 3$). The variance distribution $CV(\lambda)$ is shown for varying λ in Figure A.2a. The optimal λ leading to the minimal variance has been selected to compute the approximation of the sin signal shown in Figure A.2b.

In Figure A.2a thin lines indicate the sample variance. When (A.1) is considered as the mean μ_s and a sample is $s_i = \frac{y_i - f_\lambda(x_i)}{1-S_{ii}}$, then the sample variance is calculated as

$$\mathrm{Var} = \frac{1}{N}\Sigma_i^N (\mu_s - s_i)^2 . \tag{A.2}$$

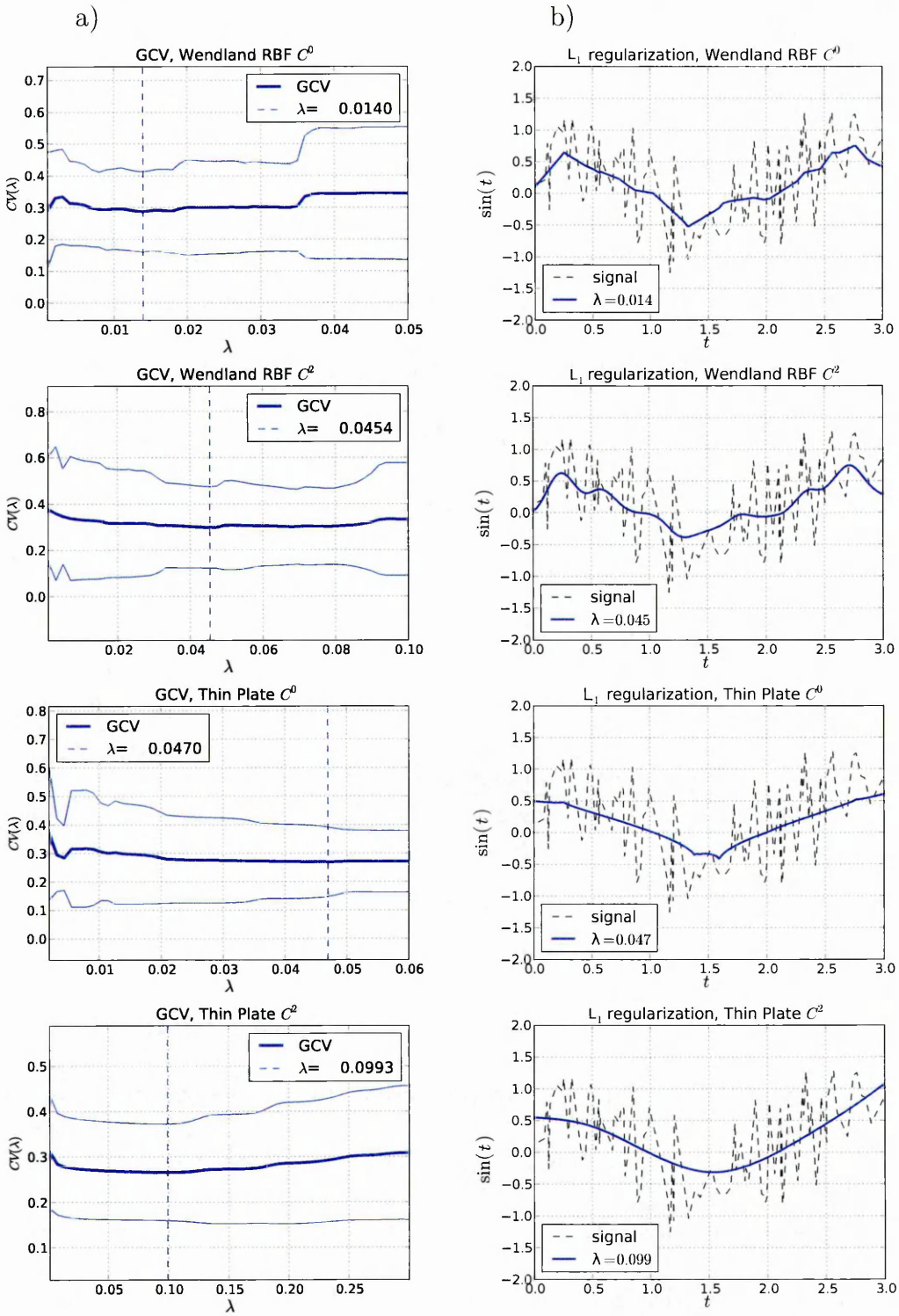


Figure A.3: a) The $CV(\lambda)$ estimation. b) $\sin(t)$ approximation with optimal λ.

Note that to simplify matters the sample variance is computed assuming a normal distribution. In contrast to the correct computation using χ^2 statistics, the applied

computation is more conservative and gives larger values for (A.2) as also pointed out by Härdle and Simar (2012). This is, however, not critical here since the results are only considered in the visualization.

Theoretical studies of GCV for Tikhonov regression (Wahba, 1990) justify that it converges asymptotically. However, good results cannot be expected for small datasets. Selecting $N \geq 100$ samples for each λ_m has been identified as reliable in evaluation experiments. This is also the number proposed by Bach (2008) for general datasets. Even though the GCV method is much more efficient than general CV approaches, it requires solving the full regression system several times. For large systems this might be inappropriate. Today, regularization tasks with RBF are in the focus of mathematical research. Lin and Yuan (2006) proposed a simple formula computing optimal λ from the number of samples and the differential degree of the approximation:

$$\lambda \approx N^{-(2m+2)/3}.$$

However, it is not explained how this formula has been established. This approximately agrees with the GCV-estimated values in the presented experiments shown in Figure A.3. Though, more research on this topic is required, which is unfortunately outside of the scope of this work. The interested reader is referred to the recent investigations presented by Zhang et al. (2011) and Friedman (2012), who seeks to partly integrate the cross validation into the optimization processes.

Appendix B: L_1 Subgradients

Chapter 4 introduced the ADMM algorithm. In its second step, it relies on the minimization of the Lagrangian (4.17) with respect to $\mathbf{z}$. The minimization

$$\min_{\mathbf{z}} g(\mathbf{z}) = \lambda \parallel z \parallel_1 + \mathbf{b}^T(D\boldsymbol{\alpha} - \mathbf{z}) + \frac{\rho}{2}(\boldsymbol{\alpha}^T D^T D\boldsymbol{\alpha} - 2\mathbf{z}^T D\boldsymbol{\alpha} + \mathbf{z}^T\mathbf{z})$$

is carried out via differentiation for each element of z_i independently:

$$z_i = \frac{\mathbf{b}_i}{\rho} + (D\boldsymbol{\alpha})_i - \frac{\lambda}{\rho}\text{sign}(z_i).$$

This is further simplified to $z_i = a - \lambda/\rho\text{sign}(z_i)$. By induction, the following possible solutions are:

- When $z_i > 0$, then $a > \lambda/\rho$, thus $z_i = a - \lambda/\rho$.
- When $z_i < 0$, then $a < -\lambda/\rho$, thus $z_i = a + \lambda/\rho$.

This leads to a simplified notation:

$$z_i^{k+1} = \begin{cases} 0 & \text{if } |a| < \frac{\lambda}{\rho} \\ \frac{b_i^k}{\rho} + (D\boldsymbol{\alpha}^k)_i - \frac{\lambda}{\rho}\text{sign}(z_i^k) & \text{else.} \end{cases}$$

The form above is the extension of the *Shrinkage-Operator* from Tibshirani (1994).

Appendix C: Partial Derivatives of the RBF

Table C.1: Gradients for RBFs, for 3D data with $d = 3$

RBF Type	$\nabla_{x_m}\varphi(\mathbf{x}_m, \mathbf{x}_i)$
Thin-Plate $\varphi(r) = (\frac{r}{\sigma})^{2m-d}$	$(\mathbf{x}_i - \mathbf{x}_m)(2m-d)\sigma^{d-2m}r^{2m-d-2}$
CSRBF C^0: $\varphi(r) = (1-r)_+^2$ C^2: $\varphi(r) = (1-r)_+^4(4r+1)$	$\frac{2}{r}(\mathbf{x}_i - \mathbf{x}_m)(1-r)_+$ $\frac{4}{r}(\mathbf{x}_i - \mathbf{x}_m)\left((1-r)_+^4 - (1-r)_+^3(4r+1)\right)$
C^4: $\varphi(r) = (1-r)_+^6(35r^2+18r+3)$	$\frac{6}{r}(\mathbf{x}_i - \mathbf{x}_m)(1-r)_+^5($ $(1-r)(70r+18) - (35r^2+18r+3))$
Gaussian $\varphi(r) = e^{-r^2}$	$-2(\mathbf{x}_i - \mathbf{x}_m)e^{-r^2}$

Table C.2: Second order derivatives with respect to radius r for $d = 3$

RBF Type	$\partial_{rr}\varphi(r)$
Thin-Plate $\varphi(r) = r^{2m-d}$	$(2m-4)\ (2m-3)\ r^{2m-5}$
CSRBF C^0: $\varphi(r) = (1-r)_+^2$ C^2: $\varphi(r) = (1-r)_+^4(4r+1)$ C^4: $\varphi(r) = (1-r)_+^6(35r^2+18r+3)$	-2 if $r < 1$, otherwise 0. $12\,(1-r)_+^2\,(4r+1) - 32(1-r)_+^3$ $30(1-r)_+^4(35r^2+18r+3) -$ $12(1-r)_+^5(70r+18) + 70(1-r)_+^6$
Gaussian $\varphi(r) = e^{-r^2}$	$4r^2e^{-t^2} - 2e^{-t^2}$

Appendix D: Parameter Selection for Poisson and SSDF

These experiments aim at identifying the proper depth parameter for the Poisson and SSDF algorithms. In both the octree depth parameter needs to be selected by the user beforehand. The evaluation considers the median of the shape distance error and the median of the curvature indicating over-fitting. For the *Facade* dataset depth = 8 has been identified as the best depth for both algorithms. As for the *Kitchen* dataset depth = 7 is selected for both methods. These values are further employed in all benchmark experiments presented in Chapter 7.

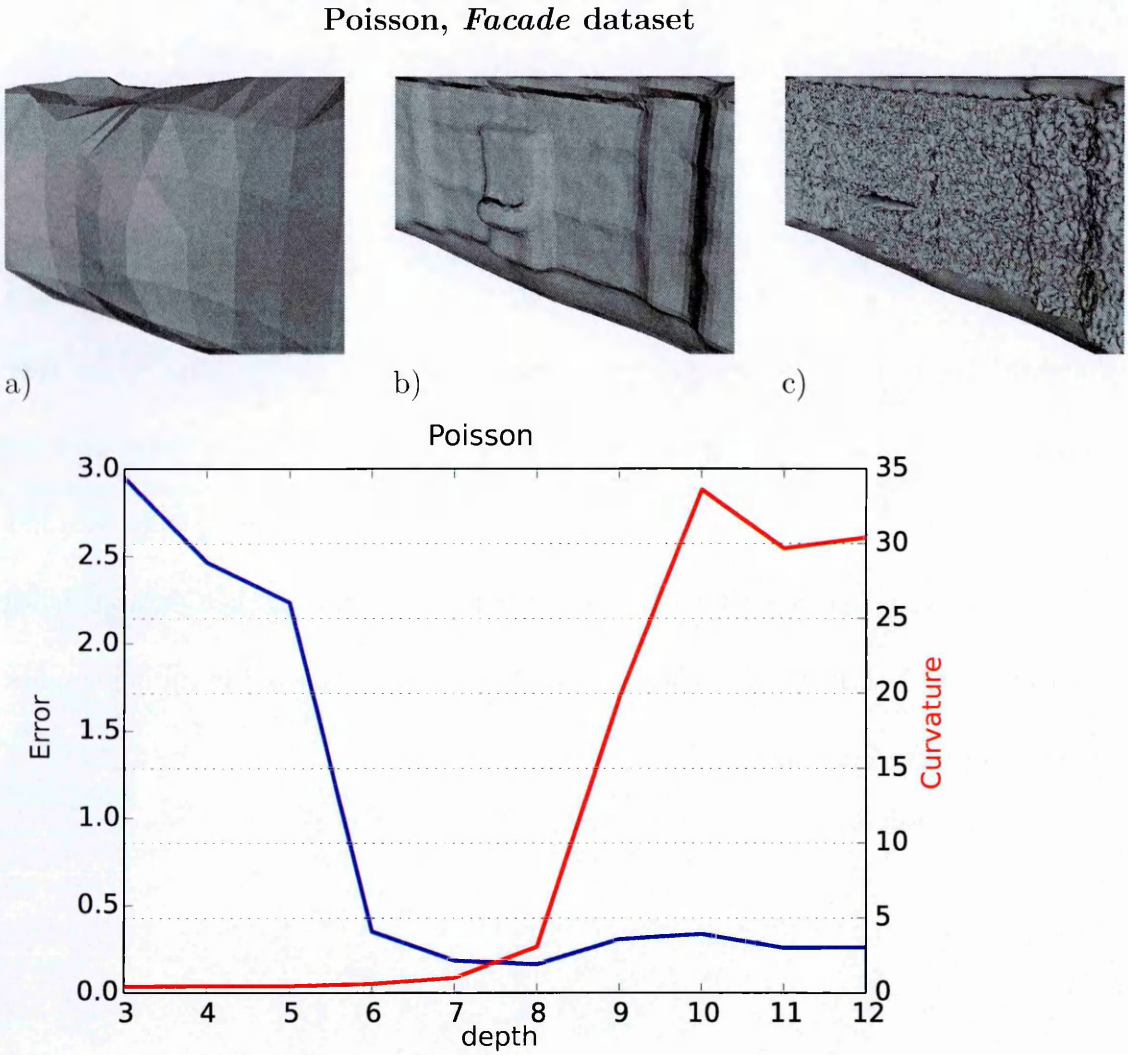


Figure D.1: a) $depth = 4$, b) $depth = 8$, c) $depth = 12$, d) approximation error and curvature of the Poisson reconstruction.

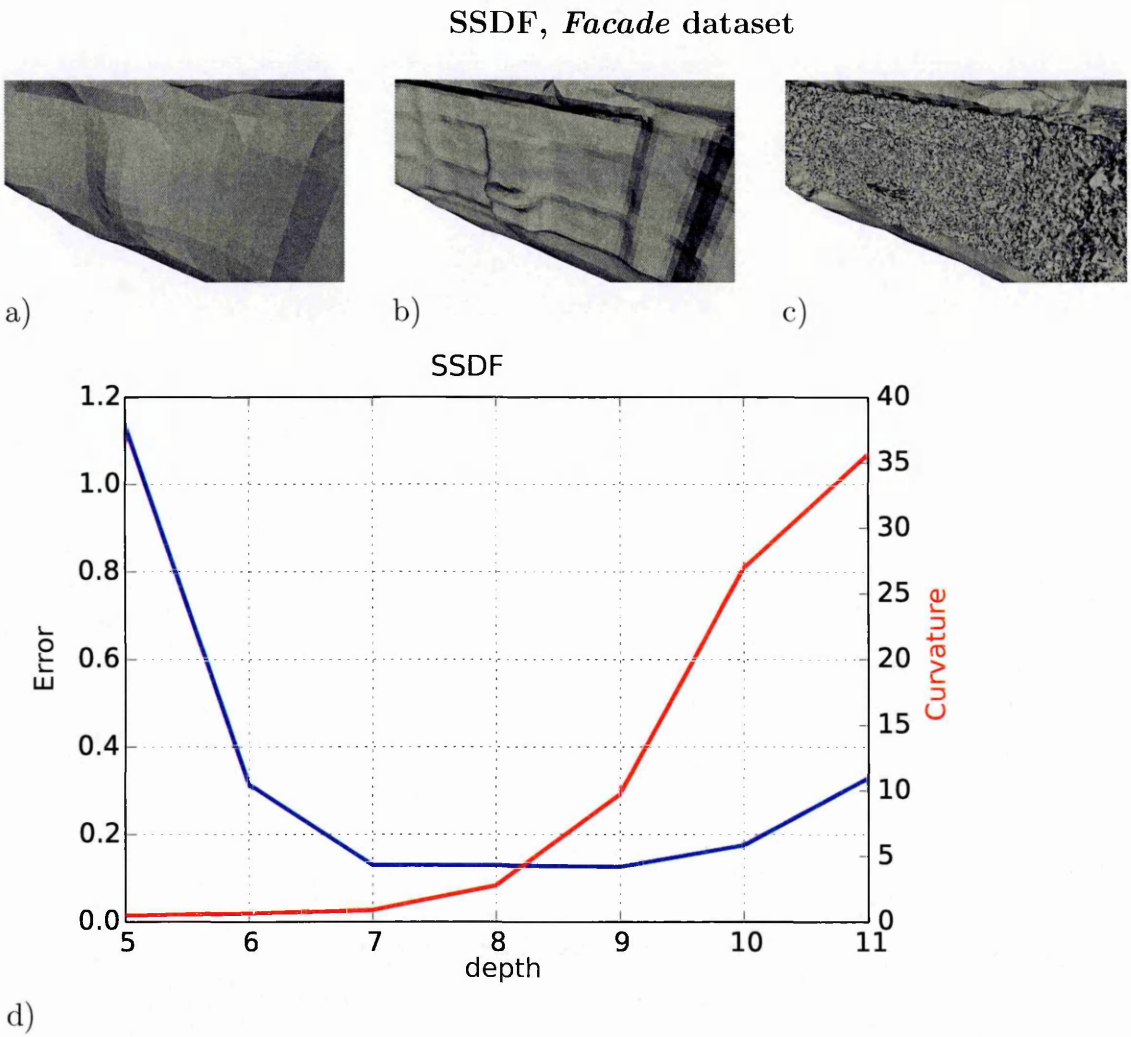
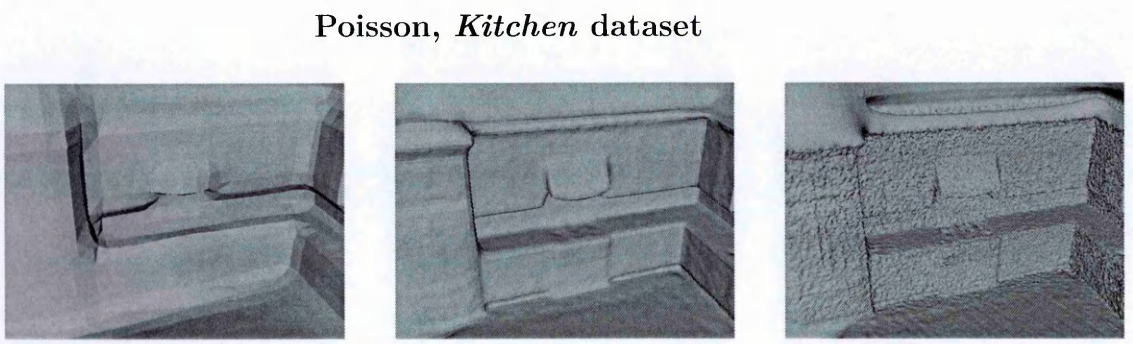
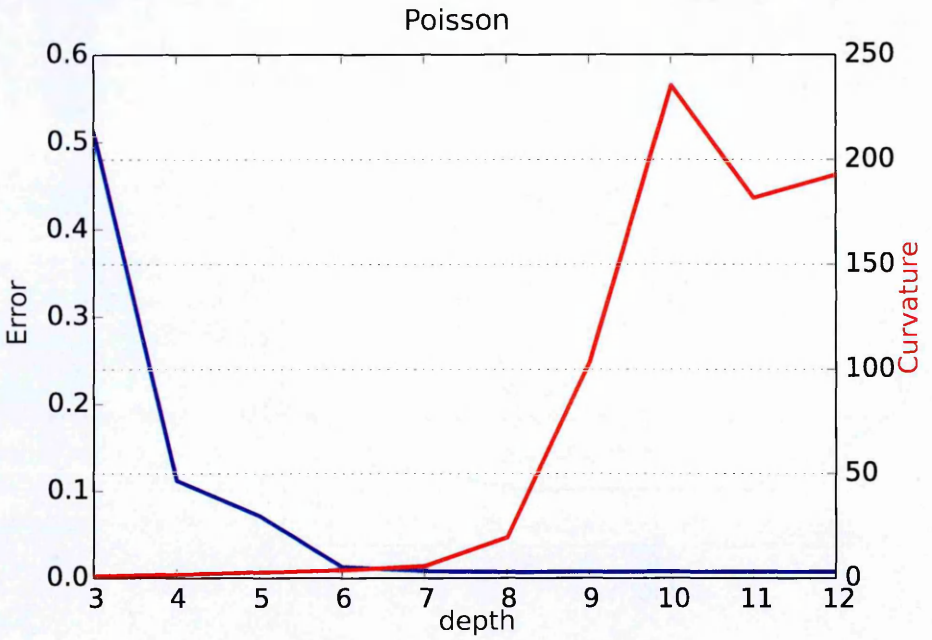


Figure D.2: a) $depth = 4$, b) $depth = 8$, c) $depth = 12$, d) approximation error and curvature of the SSDF reconstruction.



a) b) c)



d)

Figure D.3: a) $depth = 4$, b) $depth = 7$, c) $depth = 12$, d) error and curvature of the Poisson reconstruction.

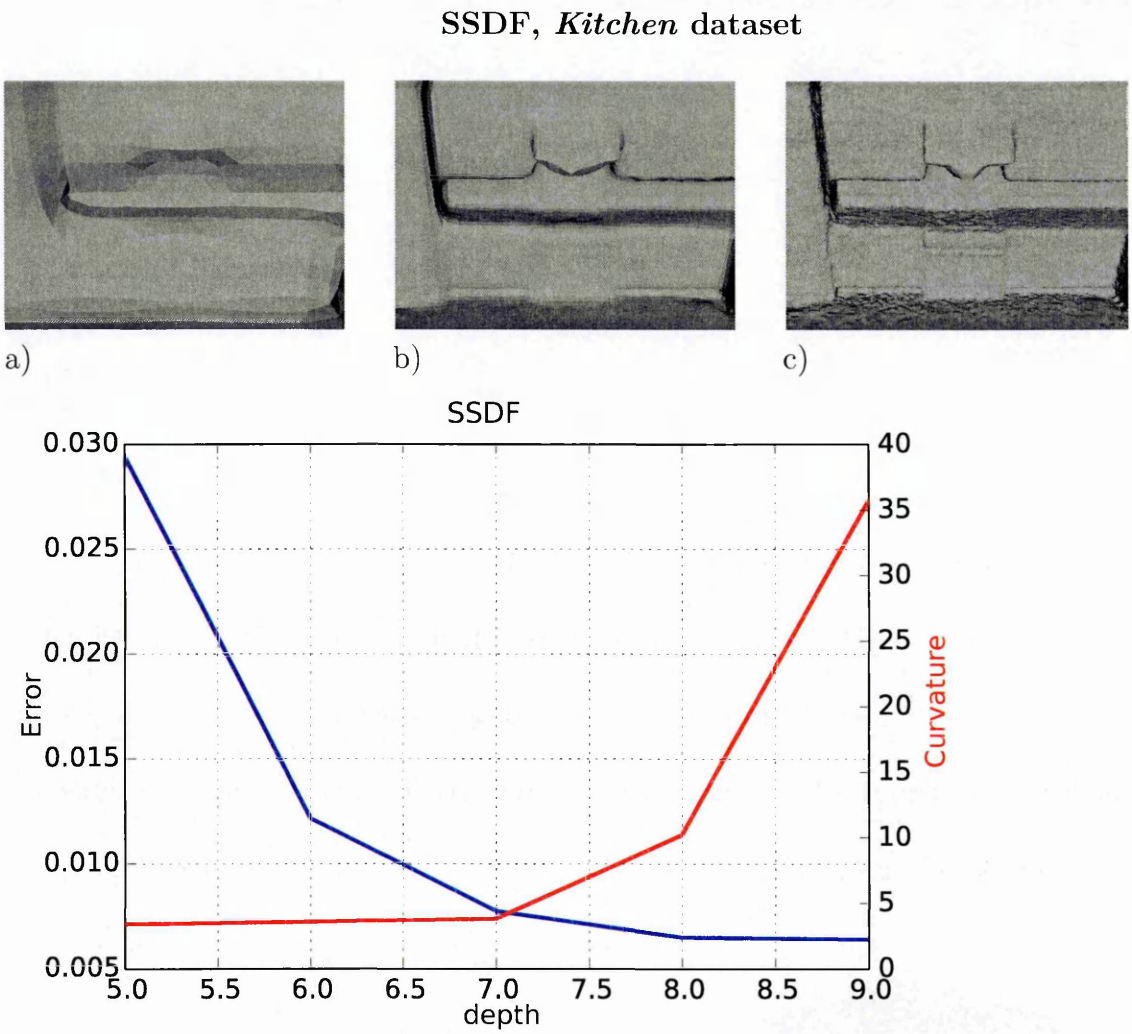
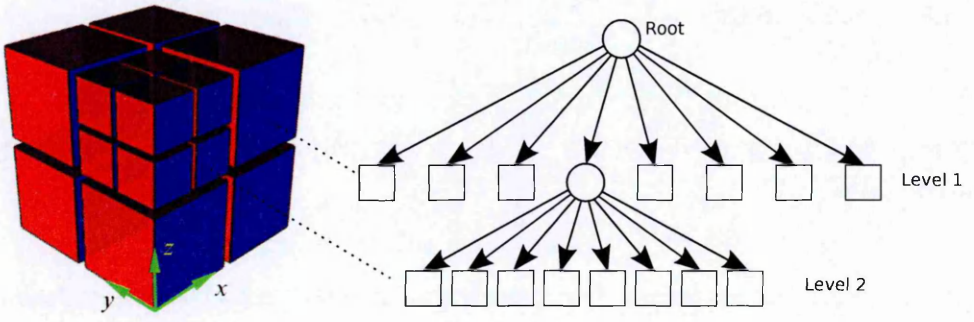


Figure D.4: a) $depth = 4$, b) $depth = 7$, c) $depth = 12$, d) error and curvature of the SSDF reconstruction.

Appendix E: Data-adaptive RBF distribution

E.1 Octree

Given an unstructured list of 3D points, computing $f(\mathbf{x})$ from the neighbours around $\mathbf{x}$ usually requires iterating through all N samples. This process is very time-consuming. Thus, the samples are structured with an octree which allows to access neighbours of a point much faster than iterating the full dataset. An octree is a spatial hierarchical grid that divides the 3D space into cubes of equivalent volumes. Beginning with the root node which encloses the full data volume, the second level contains eight blocks of smaller sizes. The third level can also contain further branches until the expected level of detail is achieved. When a leaf of the octree is accessed by a 3D coordinate, the



a) b)

Figure E.1: Exemplary space partitioning with an octree. a) The volume partitions of two different levels, b) hierarchical logical decomposition.

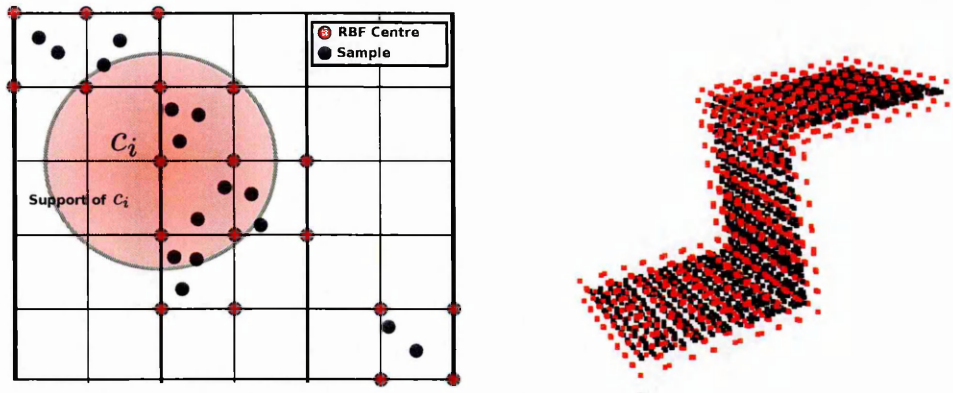


Figure E.2: The RBF centres (red dots) are placed on a grid around the samples (black dots). a) 2D, b) 3D illustration.

complexity is $\mathcal{O}(d)$ with d as the depth of the octree. Thus, the level of detail directly affects the data access speed and consequently the computation time of $f(\mathbf{x})$. The octree is a fundamental part when it comes to estimating the RBF sizes from the data.

E.2 Data-Adaptive RBF

The RBF centres are distributed uniformly between the data samples before the shape reconstruction process can start to optimize the values of $\boldsymbol{\alpha}$.

Initially, the full dataset is structured in an octree to enable faster data grouping and access. The depth parameter is estimated automatically and is controlled by the required number of included 3D points. The subdivision process is terminated as soon as an octree node has less samples than required by a predefined parameter n_c.

Each octree node stores the 3D samples and the corresponding mean position as shown in Figure E.2. RBF centres are placed on octree leaf corners around each cube before the approximation process starts. Figure E.2a illustrates the distribution of the RBFs on a 2D grid and when applied on 3D samples of the step function. This strategy

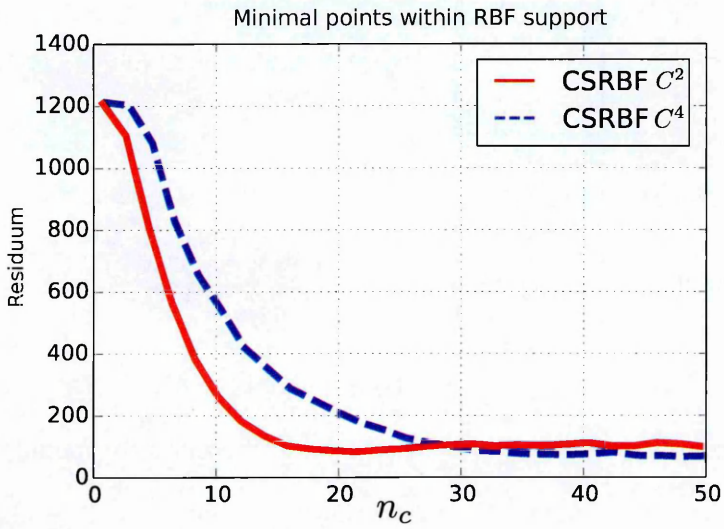


Figure E.3: The effect of n_c on the ADMM-GS residuum.

allows to adopt the framework to the data without user interaction. Initially, similar approaches have been discussed in physical numerics, today also known as *mesh free* methods (Liu, 2002).

When distributing the RBF centres on the adaptive grid, the support size s_i is a critical parameter that influences the achieved approximation quality. When an RBF is required to enclose n_c samples with r_{max} as the distance to the furthermost sample, the support s_i is set to $s_i = r_{max}$. In Chapter 4 the theoretical optimal scaling of RBF has been identified as $s/q_x > 5$. This means that when a planar shape is approximated, the number of included samples should be $5 \times 5 = 25 < n_c$. This is validated by experiments shown in Figure E.3, where the optimal number of included samples has been identified as $n_c > 25$. Larger n_c values may improve the approximation quality but definitely increase the probability of over-smoothing and also raise the computation time. Thus, the value should be selected as small as possible. Based on these considerations, $n_c = 30$ has been selected for all experiments.

References

Agoston, M. (2005). *Computer Graphics and Geometric Modelling: Implementation & Algorithms.* Computer Graphics and Geometric Modeling. Springer.

Alexa, M., J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva (2001). Point set surfaces. In *Proceedings of the Conference on Visualization '01*, VIS '01, Washington, DC, USA, pp. 21–28. IEEE Computer Society.

Allen, D. M. (1974). The Relationship between Variable Selection and Data Augmentation and a Method for Prediction. *Technometrics 16*(1).

Andres, B., B. T, and J. H. Kappes (2012). OpenGM: A c++ library for discrete graphical models. *ArXiv e-prints.*

Arikan, M., M. Schwärzler, S. Flöry, M. Wimmer, and S. Maierhofer (2013, January). O-snap: Optimization-based snapping for modeling architecture. *ACM Transactions on Graphics 32*(1), 6:1–6:15.

Avron, H., A. Sharf, C. Greif, and D. Cohen-Or (2010, November). l_1 sparse reconstruction of sharp point set surfaces. *ACM Trans. Graph. 29*(5), 135:1–135:12.

Bach, F., R. Jenatton, and J. Mairal (2011). *Optimization with Sparsity-Inducing Penalties (Foundations and Trends(R) in Machine Learning)*. Hanover, MA, USA: Now Publishers Inc.

Bach, F. R. (2008). Bolasso: model consistent lasso estimation through the bootstrap. In W. W. Cohen, A. McCallum, and S. T. Roweis (Eds.), *ICML*, Volume 307 of *ACM International Conference Proceeding Series*, pp. 33–40. ACM.

Bach, F. R., R. Jenatton, J. Mairal, and G. Obozinski (2012). Optimization with sparsity-inducing penalties. *Foundations and Trends in Machine Learning 4*(1), 1–106.

Bajaj, C. L., F. Bernardini, and G. Xu (1995). Automatic reconstruction of surfaces and scalar fields from 3d scans. In *Proceedings of the 22Nd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '95, New York, NY, USA, pp. 109–118. ACM.

Bao, Y., M. Chandraker, Y. Lin, and S. Savarese (2013). Dense object reconstruction using semantic priors. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*.

Berger, M., J. A. Levine, L. G. Nonato, G. Taubin, and C. T. Silva (2013). A benchmark for surface reconstruction. *ACM Trans. Graph. 32*(2), 20.

Bernardini, F., J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin (1999, October). The ball-pivoting algorithm for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics 5*(4), 349–359.

Berner, A., M. Wand, N. J. Mitra, D. Mewes, and H.-P. Seidel (2011). Shape analysis with subspace symmetries. *Computer Graphics Forum 30*(2). Proceedings Eurographics 2011.

Bickel, J. (2007). Discussion of "the dantzig selector: statistical estimation when p is much larger than n", by candes and tao. *Annals of Statistics*, 2352–2357.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc.

Blender Online Community (2016). *Blender - a 3D modelling and rendering package*. Blender Institute, Amsterdam: Blender Foundation.

Bodenmüller, T. (2009). *Streaming surface reconstruction from real time 3D-measurements*. Ph. D. thesis, Technical University Munich.

Boulch, A., M. de La Gorce, and R. Marlet (2014). Piecewise-planar 3d reconstruction with edge and corner regularization. *Computer Graphics Forum 33*(5), 55–64.

Bowyer, A. (1981). Computing dirichlet tessellations. *The Computer Journal 24*(2), 162–166.

Boyd, S., N. Parikh, E. Chu, B. Peleato, and J. Eckstein (2011, January). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn. 3*(1), 1–122.

Boyd, S. and L. Vandenberghe (2004). *Convex Optimization*. Cambridge University Press.

Boykov, Y., O. Veksler, and R. Zabih (1999). Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence 23*, 2001.

Bredies, K., K. Kunisch, and T. Pock (2010, September). Total generalized variation. *SIAM J. Img. Sci. 3*(3), 492–526.

Bridson, R., S. Marino, and R. Fedkiw (2003). Simulation of clothing with folds and wrinkles. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, Aire-la-Ville, Switzerland, Switzerland, pp. 28–36. Eurographics Association.

Calakli, F. and G. Taubin (2011). Ssd: Smooth signed distance surface reconstruction. *Computer Graphics Forum 30*(7), 1993–2002.

Canelhas, D. R., T. Stoyanov, and A. J. Lilienthal (2013). Sdf tracker: A parallel algorithm for on-line pose estimation and scene reconstruction from depth images. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 3671–3676. IEEE.

Carr, J. C., R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans (2001). Reconstruction and representation of 3d objects with radial basis functions. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, New York, NY, USA, pp. 67–76. ACM.

Cazals, F. and J. Giesen (2004, November). Delaunay Triangulation Based Surface Reconstruction: Ideas and Algorithms. Technical Report RR-5393.

Chen, S. S., D. L. Donoho, Michael, and A. Saunders (1998). Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing 20*, 33–61.

Duchon, J. (1977). Splines minimizing rotation-invariant semi-norms in sobolev spaces. In W. Schempp and K. Zeller (Eds.), *Constructive Theory of Functions of Several Variables*, Volume 571 of *Lecture Notes in Mathematics*, pp. 85–100. Springer Berlin Heidelberg.

Edelsbrunner, H. (1987). *Algorithms in Combinatorial Geometry.* European Association for Theoretical Computer Science: EATCS monographs on theoretical computer science. Springer.

Edelsbrunner, H. and E. P. Mücke (1994, January). Three-dimensional alpha shapes. *ACM Trans. Graph. 13*(1), 43–72.

Elad, M. (2010). *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing* (1st ed.). Springer Publishing Company, Incorporated.

Fasshauer, G. E. (2011). Positive definite kernels: Past, present and future.

Fleishman, S., D. Cohen-Or, and C. T. Silva (2005, July). Robust moving least-squares fitting with sharp features. *ACM Trans. Graph. 24*(3), 544–552.

Floater, M. and K. Hormann (2005). Surface parameterization: a tutorial and survey. In N. Dodgson, M. Floater, and M. Sabin (Eds.), *Advances in Multiresolution for Geometric Modelling*, Mathematics and Visualization, pp. 157–186. Springer Berlin Heidelberg.

Friedman, J. H. (2012). Fast sparse regression and classification. *International Journal of Forecasting 28*(3), 722 – 738.

Funk, E. and A. Börner (2014). Large scale 3d modelling via sparse volumes. In *Proc. of the Workshop on 3D Modelling, 3D-Nord-Ost, Berlin.*

Funk, E., L. S. Dooley, and A. Boerner (2015). Tvl1 shape approximation from scattered 3d data. In *Proceedings of the 10th International Conference on Computer Vision Theory and Applications (VISIGRAPP 2015)*, pp. 294–304.

Funk, E., L. S. Dooley, and A. Börner (2016). TVL_1 Planarity Regularization for 3D Shape Approximation. *Communications in Computer and Information Science 598*, 274–294.

Funk, E., L. S. Dooley, and A. Börner (2014). Implicit scene modelling from imprecise point clouds. In *Proc. of the International Conference on 3D Indoor Modelling, International Society International Society for Photogrammetry and Remote Sensing, Kape Town, South Afrika.*

Funk, E., L. S. Dooley, S. Zuev, and A. Boerner (2012). Visual recognition of scenes and objects on trains using stereo cameras. *European Navigation Conference.*

Funk, E., D. Griessbach, D. Baumbach, I. Ernst, A. Boerner, and S. Zuev (2011). Segmentation of large point-clouds using recursive local pca. *Indoor Positioning and Navigation (IPIN).*

Gobbetti, E., D. Kasik, and S.-e. Yoon (2008). Technical strategies for massive model visualization. In *Proceedings of the 2008 ACM Symposium on Solid and Physical Modeling*, SPM '08, New York, NY, USA, pp. 405–415. ACM.

Goldstein, T. and S. Osher (2009, April). The split bregman method for l1-regularized problems. *SIAM J. Img. Sci. 2*(2), 323–343.

Gomes, A., I. Voiculescu, J. Jorge, B. Wyvill, and C. Galbraith (2009). *Implicit Curves and Surfaces: Mathematics, Data Structures and Algorithms* (1st ed.). Springer Publishing Company, Incorporated.

Gonin, R. (1989). *Nonlinear Lp-Norm Estimation.* Statistics: A Series of Textbooks and Monographs. Taylor & Francis.

Goodfellow, I., A. Courville, and Y. Bengio (2015). Deep learning. Book in preparation for MIT Press.

Google Inc. (2014). 3d warehouse. `https://3dwarehouse.sketchup.com/`. Accessed: 2014-06-6.

Greenwood, D. C. (2007). Reliability of journal impact factor rankings. *BMC Medical Research Methodology 7*, 48+.

Grießbach, D., D. Baumbach, A. Börner, M. Buder, I. Ernst, E. Funk, J. Wohlfeil, and S. Zuev (2012). Ips - a system for real-time navigation and 3d modeling. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XXXIX-B5*, 21–26.

Griessbach, D., D. Baumbach, and S. Zuev (2014). Stereo-vision-aided inertial navigation for unknown indoor and outdoor environments. *Internation Conference in Indoor Naviation, IEEE.*

Guendelman, E., R. Bridson, and R. Fedkiw (2003, July). Nonconvex rigid bodies with stacking. *ACM Trans. Graph. 22*(3), 871–878.

Guennebaud, G. and M. Gross (2007, July). Algebraic point set surfaces. *ACM Trans. Graph. 26*(3).

Härdle, W. and L. Simar (2012). *Applied Multivariate Statistical Analysis.* Springer.

Hardy, R. L. (1971). Multiquadric equations of topography and other irregular surfaces.

Harrell, F. (2015). *Regression Modeling Strategies: With Applications to Linear Models, Logistic Regression, and Survival Analysis.* Graduate Texts in Mathematics. Springer.

Hastie, T., R. Tibshirani, and J. Friedman (2008). *The elements of statistical learning: data mining, inference and prediction* (2 ed.). Springer.

Heroux, M. A., R. A. Bartlett, V. E. Howle, R. J. Hoekstra, J. J. Hu, T. G. Kolda, R. B. Lehoucq, K. R. Long, R. P. Pawlowski, E. T. Phipps, A. G. Salinger, H. K. Thornquist, R. S. Tuminaro, J. M. Willenbring, A. Williams, and K. S. Stanley (2005, September). An overview of the trilinos project. *ACM Trans. Math. Softw. 31*(3), 397–423.

Hestenes, M. R. (1969). Multiplier and gradient methods. *Journal of Optimization Theory and Applications 4*(5), 303–320.

Hirschmüller, H. (2011, September). Semi-global matching - motivation, developments and applications. In D. Fritsch (Ed.), *Photogrammetric Week*, pp. 173–184. Wichmann.

Hoff, III, K. E., J. Keyser, M. Lin, D. Manocha, and T. Culver (1999). Fast computation of generalized voronoi diagrams using graphics hardware. In *Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '99, New York, NY, USA, pp. 277–286. ACM Press/Addison-Wesley Publishing Co.

Hornung, A. and L. Kobbelt (2006). Robust reconstruction of watertight 3d models from non-uniformly sampled point clouds without normal information. In *Symposium on Geometry Processing'06*, pp. 41–50.

Huber, P., J. Wiley, and W. InterScience (1981). *Robust statistics*. Wiley New York.

Hughes, J., J. Foley, A. van Dam, and S. Feiner (2014). *Computer Graphics: Principles and Practice*. The systems programming series. Addison-Wesley.

Hägele, M. (2011). Wirtschaftlichkeitsanalysen neuartiger servicerobotik-anwendungen und ihre bedeutung für die robotik-entwicklung.

Kahan, W. (1958). *Gauss-Seidel Methods for Solving Large Systems of Linear Equations*. Ph. D. thesis, Toronto, Canada; University of Toronto.

Kazhdan, M. (2005). Reconstruction of solid models from oriented point sets. In *Proceedings of the Third Eurographics Symposium on Geometry Processing*, SGP '05, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

Kazhdan, M., M. Bolitho, and H. Hoppe (2006). Poisson surface reconstruction. In *Proceedings of the Fourth Eurographics Symposium on Geometry Processing*, SGP '06, Aire-la-Ville, Switzerland, Switzerland, pp. 61–70. Eurographics Association.

Kazhdan, M. and H. Hoppe (2013, July). Screened poisson surface reconstruction. *ACM Trans. Graph. 32*(3), 29:1–29:13.

Kolluri, R. (2005, August). Provably good moving least squares. In *Proceedings of ACM-SIAM Symposium on Discrete Algorithms*, pp. 1008–1018.

Kopetz, H. (2011). *Real-Time Systems: Design Principles for Distributed Embedded Applications*. Real-Time Systems Series. Springer.

Lai, M. (2015). Deep learning for medical image segmentation. *arXiv preprint arXiv:1505.02000*.

Lang, J.-P. (2015). Redmine. `http://redmine.org`. Accessed: 2015-04-1.

Limper, M., M. Thöner, J. Behr, and D. W. Fellner (2014). Src - a streamable format for generalized web-based 3d data transmission. In *Proceedings of the 19th International ACM Conference on 3D Web Technologies*, Web3D '14, New York, NY, USA, pp. 35–43. ACM.

Lin, Y. and M. Yuan (2006). Convergence rates of compactly supported radial basis function regularization.

Liu, G. (2002). *Mesh Free Methods: Moving Beyond the Finite Element Method*. CRC Press.

Lorensen, W. E. and H. E. Cline (1987). Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '87, New York, NY, USA, pp. 163–169. ACM.

Madych, W. R. and S. A. Nelson (1990). Multivariate Interpolation and Conditionally Positive Definite Functions. II. *Mathematics of Computation 54*(189), 211–230.

Manson, J., G. Petrova, and S. Schaefer (2008). Streaming surface reconstruction using wavelets. *Computer Graphics Forum (Proceedings of the Symposium on Geometry Processing) 27*(5), 1411–1420.

Meinshausen, N. and P. Bühlmann (2010). Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 72*(4), 417–473.

Meister, S., B. Jähne, and D. Kondermann (2012). Outdoor stereo camera system for the generation of real-world benchmark data sets. *Optical Engineering 51*(02), 021107.

Microsoft Inc. (2015). Kinect: Open source scientific tools for Python. `http://www.microsoft.com/en-us/kinectforwindows/`. Accessed: 2015-04-1.

Mosteller, F. and D. L. Wallace (1963). Inference in an authorship problem. *Journal of the American Statistical Association 58*(302), 275–309.

Muja, M. and D. G. Lowe (2014). Scalable nearest neighbor algorithms for high dimensional data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 36*.

Newcombe, R. A., S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. W. Fitzgibbon (2011). Kinectfusion: Real-time dense surface mapping and tracking. In *ISMAR*, pp. 127–136. IEEE.

Ohtake, Y., A. Belyaev, M. Alexa, G. Turk, and H.-P. Seidel (2003). Multi-level partition of unity implicits. *ACM Trans. Graph. 22*(3), 463–470.

Oztireli, C., G. Guennebaud, and M. Gross (2009). Feature Preserving Point Set Surfaces based on Non-Linear Kernel Regression. *Computer Graphics Forum 28*(2), 493–501.

Pauly, M., N. J. Mitra, J. Wallner, H. Pottmann, and L. Guibas (2008). Discovering structural regularity in 3D geometry. *ACM Transactions on Graphics 27*(3), #43, 1–11.

Piegl, L. and W. Tiller (1997). *The NURBS Book.* Monographs in Visual Communication. U.S. Government Printing Office.

Pratt, V. (1987, August). Direct least-squares fitting of algebraic surfaces. *SIGGRAPH Comput. Graph. 21*(4), 145–152.

Reinschke, K. (2014). *Lineare Regelungs- und Steuerungstheorie.* Springer Berlin Heidelberg.

Riesz, F. (1910). Untersuchungen über systeme integrierbarer funktionen. *Mathematische Annalen 69*(4), 449–497.

Robinet, F., R. Arnaud, T. Parisi, and P. Cozzi (2014). gltf: Designing an open-standard runtime asset format. In W. Engel (Ed.), *GPU Pro 5*, pp. 375–392. CRC Press.

Rogers, D. F. (2001). Preface. In D. F. Rogers (Ed.), *An Introduction to {NURBS}*, The Morgan Kaufmann Series in Computer Graphics, pp. xv – xvii. San Francisco: Morgan Kaufmann.

Roy Rosenzweig Center (2015). Zotero. `http://zotero.org`. Accessed: 2015-04-1.

Rusinkiewicz, S. and M. Levoy (2000, July). QSplat: A multiresolution point rendering system for large meshes. In *Proceedings of ACM SIGGRAPH 2000*, pp. 343–352.

Rusu, R. B. and S. Cousins (2011, May 9-13). 3D is here: Point Cloud Library (PCL). In *IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China.

Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems* (2nd ed.). Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.

Scholkopf, B. and A. J. Smola (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* Cambridge, MA, USA: MIT Press.

Schreiner, J., A. Asirvatham, E. Praun, and H. Hoppe (2004, August). Inter-surface mapping. *ACM Trans. Graph. 23*(3), 870–877.

Schwaber, K. (2004). *Agile Project Management With Scrum.* Redmond, WA, USA: Microsoft Press.

Seglen, P. O. (1997). Why the impact factor of journals should not be used for evaluating research. *BMJ 314*(7079), 497.

Sharf, A., T. Lewiner, G. Shklarski, S. Toledo, and D. Cohen-Or (2007, august). Interactive topology-aware surface reconstruction. In *Siggraph 2007 (ACM Transaction on Graphics)*, Volume 26, San Diego, pp. 43.1–43.9. ACM.

Sheffer, A., E. Praun, and K. Rose (2007). Mesh parameterization methods and their applications. *Foundations and Trends in Computer Graphics and Vision 2*(2), 105–171.

Shen, C.-H., H. Fu, K. Chen, and S.-M. Hu (2012). Structure recovery by part assembly. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH Asia 2012 31*(6), 180:1–180:11.

Smola, A. J. and B. Schölkopf (1998). On a kernel-based method for pattern recognition, regression, approximation, and operator inversion. *Algorithmica 22*(1/2), 211–231.

Strecha, C., W. von Hansen, L. J. V. Gool, P. Fua, and U. Thoennessen (2008). On benchmarking camera calibration and multi-view stereo for high resolution imagery. In *CVPR*. IEEE Computer Society.

Strekalovskiy, E., A. Chambolle, and D. Cremers (2014). Convex relaxation of vectorial problems with coupled regularization. 7(1), 294–336.

Sturm, J., N. Engelhard, F. Endres, W. Burgard, and D. Cremers (2012, Oct.). A benchmark for the evaluation of rgb-d slam systems. In *Proc. of the International Conference on Intelligent Robot Systems (IROS)*.

Takens, F. (1968). The minimal number of critical points of a function on a compact manifold and the lusternik-schnirelman category. *Inventiones mathematicae 6*(3), 197–244.

Tibshirani, R. (1994). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B 58*, 267–288.

Tikhonov, A. N. (1943). On the stability of inverse problems. *Doklady Akademii Nauk SSSR 39 (5)*, 195–198.

Tufte, E. (1990). *Envisioning Information.* Cheshire, CT, USA: Graphics Press.

Wahba, G. (1990). *Spline models for observational data*, Volume 59 of *CBMS-NSF Regional Conference Series in Applied Mathematics.* Philadelphia, PA: Society for Industrial and Applied Mathematics (SIAM).

Wahba, G. and J. Wendelberger (1980). Some New Mathematical Methods for Variational Objective Analysis Using Splines and Cross Validation. *Monthly Weather Review 108*, 1122–1143.

Walder, C., B. Schölkopf, and O. Chapelle (2006). Implicit surface modelling with a globally regularised basis of compact support. *Computer Graphics Forum 25*(3), 635–644.

Wang, Y., A. Yagola, and C. Yang (2011). *Optimization and Regularization for Computational Inverse Problems and Applications.* Springer.

Wei, J., B. Resch, and H. Lensch (2014). Multi-view depth map estimation with cross-view consistency. In *Proceedings of the British Machine Vision Conference*. BMVA Press.

Wekel, T. and O. Hellwich (2015). Semantic segmentation and model fitting for triangulated surfaces using random jumps and graph cuts. In *VISAPP 2015 - Proceedings of the 10th International Conference on Computer Vision Theory and Applications, Volume 3, Berlin, Germany, 11-14 March, 2015.*, pp. 241–250.

Wendland, H. (1995). Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in Computational Mathematics 4*(1), 389–396.

Wendland, H. (2004). *Scattered Data Approximation*. Cambridge University Press.

Wolff, D. (2013). *OpenGL 4 Shading Language Cookbook, Second Edition*. EBL-Schweitzer. Packt Publishing.

Yan, Z., V. Jagadeesh, D. Decoste, W. Di, and R. Piramuthu (2014, Oct). Hd-cnn: Hierarchical deep convolutional neural network for image classification. In *arXiv*.

Yang, J., Y. Zhang, and W. Yin (2010, APR). A Fast Alternating Direction Method for TVL1-L2 Signal Reconstruction From Partial Fourier Data. *IEEE Journal of Selected Topics in Signal Processing 4*(2), 288–297.

Zhang, F., T. Lai, B. Rajaratnam, N. Zhang, and S. U. D. of Statistics (2011). *Cross-validation and Regression Analysis in High-dimensional Sparse Linear Models*. Stanford University.